

AD-A162 389

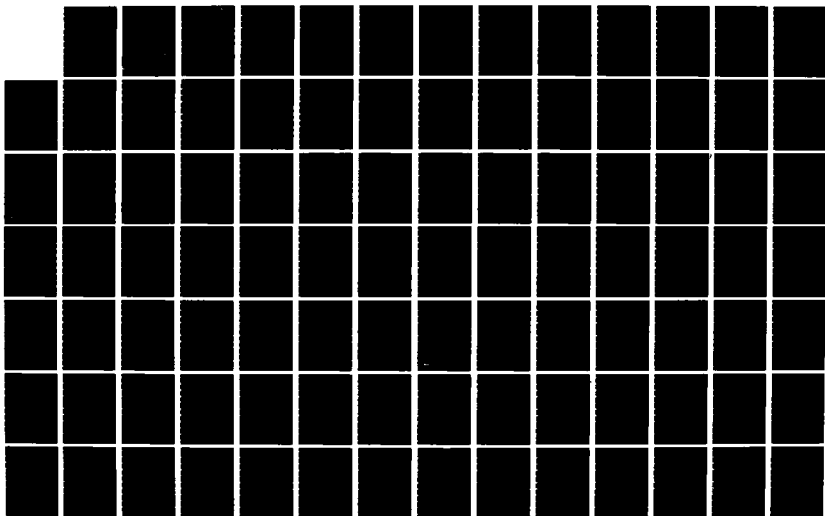
KNOWLEDGE REPRESENTATION AND NATURAL-LANGUAGE SEMANTICS
(U) SRI INTERNATIONAL MENLO PARK CA R C MOORE AUG 85
AFOSR-TR-85-1898 F49628-82-K-8831

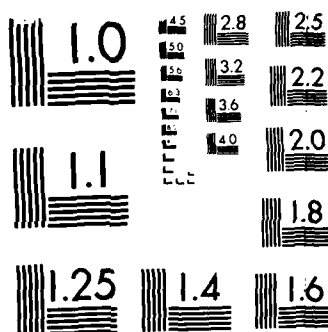
1/4

UNCLASSIFIED

F/G 5/7

ML





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

KNOWLEDGE REPRESENTATION AND NATURAL-LANGUAGE SEMANTICS

August 1985

Final Technical Report
Covering the Period June 1, 1982 to May 30, 1985

By: Robert C. Moore, Staff Scientist
Artificial Intelligence Center
Computer Science and Technology Division

Contributing Authors:

William Croft David J. Israel
Edwin Pednault Kurt Konolige

Prepared for:

Air Force Office of Scientific Research
Building 410
Bolling Air Force Base
Washington, D. C. 20332

Attention: Dr. Robert Buchal

Contract No. F49620-82-K-0031

SRI Project 4488

SRI International
333 Ravenswood Avenue
Menlo Park, California 94025-3493
Telephone: (415) 326-6200
Cable: SRI INTL MPK
TWX: 910-373-2046
Telex: 334 486

DTIC
ELECTE
DEC 11 1985
S B

DTIC FILE COPY



AD-A162 389

AD-A163389

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION Unclassified			1b. RESTRICTIVE MARKINGS	
2a. SECURITY CLASSIFICATION AUTHORITY			3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release, distribution unlimited	
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE N/A				
4. PERFORMING ORGANIZATION REPORT NUMBER(S) SRI Project ECU 4488			5. MONITORING ORGANIZATION REPORT NUMBER(S) AFOSR-TR-88-008	
6a. NAME OF PERFORMING ORGANIZATION SRI International		6b. OFFICE SYMBOL (If applicable)	7a. NAME OF MONITORING ORGANIZATION U.S. Air Force Office of Scientific Research	
6c. ADDRESS (City, State and ZIP Code) 333 Ravenswood Avenue Menlo Park, CA 94025			7b. ADDRESS (City, State and ZIP Code) Bolling Air Force Base Washington, D.C. 20332	
8a. NAME OF FUNDING/SPONSORING ORGANIZATION U.S. Air Force Office of Scientific Research		8b. OFFICE SYMBOL (If applicable) NM	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER F49620-82-K-0031	
8c. ADDRESS (City, State and ZIP Code) Bolling Air Force Base Washington, D.C. 20332			10. SOURCE OF FUNDING NOS.	
			PROGRAM ELEMENT NO. 61103F	PROJECT NO. 3304
11. TITLE (Include Security Classification) Knowledge Representation & Natural Language Semantics				
12. PERSONAL AUTHOR(S) Robert C. Moore				
13a. TYPE OF REPORT Final		13b. TIME COVERED FROM 6/1/82 To 5/30/85	14. DATE OF REPORT (Yr., Mo., Day) August 1985	
15. PAGE COUNT 354				
16. SUPPLEMENTARY NOTATION Contributing Authors: William Croft, David J. Israel, Kurt Konolige, Robert C. Moore, Edwin Pednault				
17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)	
FIELD	GROUP	SUB. GR.	Artificial Intelligence; Logic of Belief; Automatic Planning; Knowledge Representation; Logic of Knowledge and Action; Natural-Language Semantics; Nonmonotonic Logic.	
19. ABSTRACT (Continue on reverse if necessary and identify by block number) This report summarizes three years of research on a project to produce formalisms, suitable for manipulation by computer, for the representation of specific concepts that are important for natural-language semantics, and to give an independent account of the meaning of such representations using the tools of formal logic. Specific topics on which progress was made include: a logic that characterizes systems that represent and reason with information about their own beliefs, a formalism for the representation of information about the interdependence of knowledge and action, a semantical analysis of adverbial modifiers and event sentences, a formal model of belief based on deduction, additional results on the formal semantics of our logic for reasoning about one's own beliefs, a belief logic that makes weaker than usual assumptions about introspection, and a mathematically rigorous theory of plan synthesis.				
20. DISTRIBUTION/AVAILABILITY OF ABSTRACT UNCLASSIFIED/UNLIMITED <input checked="" type="checkbox"/> SAME AS RPT. <input checked="" type="checkbox"/> DTIC USERS <input type="checkbox"/>			21. ABSTRACT SECURITY CLASSIFICATION	
22a. NAME OF RESPONSIBLE INDIVIDUAL Dr. Robert Buchal			22b. TELEPHONE NUMBER (Include Area Code) (202) 761-4939	22c. OFFICE SYMBOL EQ8671 NM

CONTENTS

I	OBJECTIVES OF THE RESEARCH EFFORT	1
II	STATUS OF THE RESEARCH EFFORT	3
	A. Previous Results	3
	B. Recent Results	9
III	PUBLICATIONS	17
IV	CONFERENCE PRESENTATIONS	18
V	PERSONNEL	19
	REFERENCES	20

APPENDICES

- A SEMANTICAL CONSIDERATION ON NONMONOTONIC LOGIC
- B A FORMAL THEORY OF KNOWLEDGE AND ACTION
- C THE REPRESENTATION OF ADVERBS, ADJECTIVES AND EVENTS IN
LOGICAL FORM
- D BELIEF AND INCOMPLETENESS
- E POSSIBLE-WORLD SEMANTICS FOR AUTOEPISTEMIC LOGIC
- F A WEAK LOGIC FOR KNOWLEDGE AND BELIEF
- G PRELIMINARY REPORT ON A THEORY OF PLAN SYNTHESIS

Accession For	
NTIS GRA&I	✓
DTIC TAB	
Unannounced	
Justification	
By	
Distribution	
Availability	
Dist	Special
A-1	



AIR FORCE OVER
NO
THE
DE
MAN
Chief

I OBJECTIVES OF THE RESEARCH EFFORT

Central to almost all aspects and applications of artificial intelligence are the representation and manipulation of large bodies of knowledge about the world. When viewed from the perspective of their ability to express facts about the external world, however, most knowledge representation schemes currently used in artificial intelligence are constrained by the limits of first-order logic. That is, they provide terms for referring to individuals, predicates for expressing properties and relations of individuals, and mechanisms that achieve some of the effects of propositional connectives and quantifiers. Much research effort has been expended on ways of organizing knowledge bases and developing information retrieval mechanisms; in terms of pure expressive power, however, existing representation systems are rather limited.

This issue is brought into sharp focus when one seriously attempts to analyze the semantic content of expressions in natural language, since many types of linguistic expressions seem to require something beyond first-order logic to represent their meaning perspicuously. Specifically, natural languages have special features for dealing with a variety of concepts that are central to our commonsense understanding of the world. For instance, linguistic systems of tense and aspect are intimately connected with commonsense conceptions of time. Adverbial modification, nominalization phenomena, and categorical distinctions among verb phrases appear to depend on such notions as state, event, and process. Predicate complement constructions frequently involve concepts of "propositional attitude" such as knowledge, belief, desire, and intention. The linguistic features of singular/plural and mass/count are used to sort out individuals, collective entities, and substances. In all these cases, either it is not clear how to express these concepts

in first-order logic at all, or it is clear that they can be expressed in first-order logic only by very indirect means.

This project has undertaken a program of basic research in knowledge representation, focusing on the representation of concepts needed for the semantic analysis of natural language. The objectives of the project are to produce formalisms, suitable for manipulation by computer, for the representation of specific concepts that are important for natural-language semantics, and to give an independent account of the meaning of such representations using the tools of formal logic.

II STATUS OF THE RESEARCH EFFORT

A. Previous Results

1. Development of Autoepistemic Logic

The major technical achievement of the first year of the project was the development of a logic that characterizes systems that represent and reason with information about their own beliefs. We call this logic "autoepistemic logic." The problem of representing and reasoning with information about knowledge or beliefs of other agents has received much attention recently in artificial intelligence. Designing a system that can represent and reason about its own beliefs, however, poses some unique problems. The nature of the difficulties is suggested by an old philosophical puzzle: Why are sentences of the form "P is true, but I don't believe P" extremely odd, although sentences of the form "P is true, but he doesn't believe P" are not? Using the first person (making a statement about one's own beliefs) makes nonsense out of a sentence that is perfectly reasonable in the third person (making a statement about someone else's beliefs).

For a simple logical language for making statements about one's own beliefs, we were able to construct a very natural formal semantics and define sets of beliefs that are both sound and complete with respect to that semantics. (Roughly speaking, a set of beliefs is sound if it contains only statements that must be true whenever the premises of the set of beliefs are true, and it is complete if it contains all the statements that must be true whenever the premises of the set of beliefs are true.)

Autoepistemic logic turns out to be quite similar to logics that have been proposed to model what is called "nonmonotonic reasoning." Commonsense reasoning is "nonmonotonic" in the sense that

we often draw, on the basis of partial information, conclusions that we later retract when we are given more complete information. The following example is frequently used to illustrate the point: If we know that Tweety is a bird, we will normally assume, in the absence of evidence to the contrary, that Tweety can fly. If, however, we later learn that Tweety is a penguin, we will withdraw our prior assumption. If we try to model this in a formal system, we seem to have a situation in which a theorem P is derivable from a set of axioms A , but is not derivable from some set A' that is a superset of A . The set of theorems, therefore, does not increase monotonically with the set of axioms; hence this sort of reasoning is said to be "nonmonotonic."

Some of the most interesting recent attempts to formalize nonmonotonic reasoning are the nonmonotonic logics developed by Drew McDermott and Jon Doyle [1] [2]. These logics, however, all have peculiarities that suggest they do not quite succeed in capturing the intuitions that prompted their development. By comparing McDermott and Doyle's logics with autoepistemic logic, we have been able to diagnose the reasons for their peculiarities and show how they can be eliminated.

Our work on autoepistemic logic is described more fully, focusing on its relationship to nonmonotonic logic in an article we have recently published [3], which is reproduced as Appendix A.

2. Representing the Dependence of Action on Knowledge

One of the representational problems we have studied is the relationship between knowledge and action. Both knowledge and action are among the basic concepts that underlie many different areas of commonsense and expert knowledge, but the interaction between the two is particularly important when applying artificial intelligence techniques to planning.

Planning sequences of actions and reasoning about their effects is one of the most thoroughly studied areas within artificial intelligence, but relatively little attention has been paid to the important role that an agent's knowledge plays in planning and acting to

achieve a goal. Virtually all planning systems in artificial intelligence are designed to operate with complete knowledge of all relevant aspects of the problem domain and problem situation. Often any statement that cannot be inferred to be true is assumed to be false. In the real world, however, planning and acting must frequently be performed without complete knowledge of the situation.

This constraint imposes two additional burdens on an intelligent agent trying to act effectively. First, when the agent entertains a plan for achieving some goal, he must consider not only whether the physical prerequisites of the plan have been satisfied, but also whether he has all the information necessary to carry out the plan. Second, he must be able to reason about what he can do to obtain necessary information that he lacks. For example, to call someone on the telephone, just being physically able to dial a telephone is not sufficient; one must also know the person's telephone number. One can plan to acquire this information, however, by looking up the number in a telephone book.

Under this project, we have refined and extended our previous work on the dependence of action on knowledge [4]. Our main thesis is that the knowledge required for an action can be analyzed as a matter of knowing what action to take. An agent could know that to call Smith on the telephone he needs to dial Smith's telephone number, but still not know what to do because he does not know precisely what action dialing Smith's telephone number is. That is, he might not know whether dialing Smith's telephone number is the action of dialing 221-1111, or dialing 221-1112, or dialing 221-1113, and so on. We may assume he has a general procedure for dialing telephone numbers, but unless he knows which number to apply it to, he does not, in the relevant sense, know what to do.

In our previous work, we successfully applied this analysis to actions that are treated as nondecomposable wholes, but our treatment of complex plans was less satisfactory. To represent complex plans, we introduced concepts of sequential actions, conditional actions, and

iterated actions. Formalizing the knowledge prerequisites of these complex actions was somewhat ad hoc, however. In particular, for conditional actions ("if P is true, then do ACTION1, otherwise do ACTION2") we had to state independently the fact that, in order to carry out a conditional action, an agent must know if the condition is true.

The work performed under this project remedies this and a number of other deficiencies. The key change is to view a complex plan as a description of a sequence of actions. Then the knowledge prerequisites of complex plans can be given a treatment similar to that for simple actions, so that the agent is assumed to have sufficient knowledge to carry out a plan if he knows what sequence of actions the plan describes. The problem of conditional actions is handled automatically, because what action is described by a conditional action description depends on whether the condition is true. Hence an agent must know whether the condition is true to know what action this is. This work is presented in full in a paper by Moore [5], included as Appendix B.

3. Semantic Analysis of Adverbial Modifiers and Event Sentences

A good example of the way a careful analysis of the meaning of natural-language expressions gives us insight into the representation of commonsense knowledge is presented by our work on the adverbial modification of event sentences. Whether or not there is a fundamental semantic distinction between event sentences, such as "John went to New York," and stative sentences, such as "John was in New York," is one of the more puzzling problems in representing the meaning of expressions in ordinary English. The latter sentence can be analyzed as saying simply that a certain relation, that of location, held between John and New York at some past time. This type of analysis seems less satisfactory, though, for the former sentence. "Went" does not seem merely to express a relation the way "is in" does. Rather, it appears to describe an event, indicated by the fact that it makes sense to ask "When did it happen?" after being told "John went to New York," but not after being told "John was in New York."

One suggestion as to how event sentences might differ from stative sentences is provided by Davidson [6], who suggests that event sentences be represented as explicitly asserting the existence of the event being described. Roughly speaking, this amounts to treating "John went to New York" as if it were "There was a going of John to New York." Davidson's suggestion is intriguing, but, heretofore, there has been relatively little evidence to support it. The study of adverbial modification of event sentences conducted under this project has provided the most convincing support to date for the kind of representation of event sentences given by Davidson and has cleared up several related problems. This work is described more fully in a paper by Croft [7], included as Appendix C.

To summarize this work briefly, we have developed a unified analysis for most "-ly" adverbs and adjectives, namely, as predicates. A small class of adverbs, all indicating modality or uncertainty ("possibly," "probably," "allegedly," etc.), must be treated as modal operators over propositions, as their semantics implies: thus, "John probably ate the cookie" would be represented as PROBABLE[EAT(JOHN,COOKIE)]. The corresponding adjectival forms are interpreted, using restricted quantification notation, as modal operators over the description; thus, "any possible solution" will be (ANY X: POSSIBLE[SOLUTION(X)]).

All other adjectives and adverbs that have the property of "factivity" (viz., if the sentence with the adverb/adjective is true, then the sentence without the adverb/adjective is also true), are predicates. The presence of "-ly" is syntactically determined: if the predicate is modifying a verb or adjective instead of a noun, the "-ly" is added. The semantic difference between "adjectives" and "adverbs" is that the former are the properties of objects, the latter of events, events being represented as event variables following Davidson [6]. Thus, "John slowly entered the room" is ENTER(E,JOHN,ROOM) & SLOW(E).

There are two unusual cases, which must be accounted for. First, a sentence like "Maggie rudely spoke to the Queen" is ambiguous

between a manner reading ("The manner in which Maggie spoke to the Queen was rude") and a fact reading ("The fact that Maggie spoke to the Queen was rude"). While the first reading is represented by modification of the event variable, the second reading represents an assertion about a state of affairs, the state of affairs of the proposition "Maggie spoke to the Queen" being true, which we represent by the FACT operator. Thus the two readings are SPEAK(E,MAGGIE,QUEEN) & RUDE(E) and SPEAK(E,MAGGIE,QUEEN) & RUDE(FACT[SPEAK(E,MAGGIE,QUEEN)]) respectively. Second, adverbs of intention ("intentionally," "willingly," etc.), which display referential opacity and other intensional behavior, must be represented as predicates taking an agent and a proposition as well as an event.

All possible derivational patterns between adverbs and adjectives are found. Adverbs like "bitterly," which take an individual and an event, are derived from adjectives that take an individual and describe his emotional state. Adverbs like "slowly," which take an event only, have derived adjectives that take an individual and a role: "John ran the mile fast" vs. "John is fast (at running the mile)." Finally, for adverbs like "rudely" or "cleverly," which take an individual and an event (or FACT operator), the corresponding adjectives are identical in semantic form: in the manner reading, "John cleverly solved the problem" and "John was clever at solving the problem" are both represented as SOLVE(E, JOHN, PROBLEM) & CLEVER(E).

Adjectives and adverbs that are "gradable" (viz., can be modified by degree terms or placed in comparative constructions) will have additional arguments in the predicate structure, and that is being investigated in other work on this project. The fact that gradability applies to both adjectives and adverbs, however, is another confirmation of their underlying semantic unity.

4. The Deduction Model of Belief

Reasoning about the knowledge and beliefs of computer and human agents is assuming increasing importance in artificial

intelligence systems for natural-language understanding, planning, and knowledge representation. A natural model of belief for robot agents is the deduction model: an agent is represented as having an initial set of beliefs about the world in some internal language and a deduction process for deriving some (but not necessarily all) logical consequences of these beliefs. Because the deduction model is an explicitly computational model, it is possible to take into account limitations of an agent's resources when reasoning.

This project has provided partial support for an investigation of a Gentzen-type formalization of the deductive model of belief. Several original results have been proven. Among these are soundness and completeness theorems for a deductive belief logic, a correspondence result that relates our deduction model to competing possible-world models, and a modal analog to Herbrand's Theorem for the belief logic. Specialized techniques for automatic deduction based on resolution have been developed using this theorem.

Several other topics of knowledge and belief have been explored from the viewpoint of the deduction model, including a theory of introspection about self-beliefs, and a theory of circumscriptive ignorance, in which facts an agent doesn't know are formalized by limiting or circumscribing the information available to him. These results are presented in the Ph.D. dissertation of Konolige [8] and are summarized in a shorter paper [9], included as Appendix D.

B. Recent Results

1. Possible-World Semantics for Autoepistemic Logic

In our previous work [3] we developed a nonmonotonic logic for modeling the beliefs of ideally rational agents who reflect on their own beliefs. We called this system "autoepistemic logic." We defined a simple and intuitive semantics for autoepistemic logic, and we were able to show that the logic was both sound and complete with respect to this semantics. However, the nonconstructive character of both the logic and its semantics made it difficult to prove the existence of sets of

beliefs satisfying all the constraints of autoepistemic logic. We have recently developed an alternative, possible-world semantics for autoepistemic logic that enables us to construct finite models for autoepistemic theories and to demonstrate the existence of sound and complete autoepistemic theories that are based on given sets of premises.

The language of autoepistemic logic is that of ordinary propositional logic, augmented by a modal operator L . Formulas of the form LP are interpreted informally to mean "P is believed" or "I believe P." For example, $P \rightarrow LP$ could be interpreted as saying "If P is true, then I believe that P is true." If a set of formulas is to be interpreted as a representation of the beliefs of a rational agent, then a formula LP will be true with respect to a certain set of beliefs if and only if P is in the set. That is, the statement "I believe P" is true for a particular agent just in case he, in fact, believes P. In the original semantics we developed for autoepistemic logic, we simply stipulated that this constraint had to be met by models of autoepistemic theories. This had the effect that the specification of a model had to include a potentially infinite list of all the formulas of the form LP that were to be taken as true. The resulting lack of structure in the models made it extremely difficult to prove results concerning the models of particular autoepistemic theories.

However, it turns out that, for autoepistemic theories representing sets of beliefs satisfying certain stability conditions, we can define models that have much more structure. The conditions of interest are that (1) the set of beliefs is closed under ordinary logical consequence, (2) whenever a formula P is believed, it is believed that P is believed, and (3) whenever a formula P is not believed, it is believed that P is not believed. We have been able to show that a set of beliefs satisfying these conditions can be characterized by a set of possible worlds such that a formula is believed if it is true in every world in the set, and a formula of the form LP is true in a particular world if P is true in every world in the set.

beliefs satisfying all the constraints of autoepistemic logic. We have recently developed an alternative, possible-world semantics for autoepistemic logic that enables us to construct finite models for autoepistemic theories and to demonstrate the existence of sound and complete autoepistemic theories that are based on given sets of premises.

The language of autoepistemic logic is that of ordinary propositional logic, augmented by a modal operator L . Formulas of the form LP are interpreted informally to mean "P is believed" or "I believe P." For example, $P \rightarrow LP$ could be interpreted as saying "If P is true, then I believe that P is true." If a set of formulas is to be interpreted as a representation of the beliefs of a rational agent, then a formula LP will be true with respect to a certain set of beliefs if and only if P is in the set. That is, the statement "I believe P" is true for a particular agent just in case he, in fact, believes P. In the original semantics we developed for autoepistemic logic, we simply stipulated that this constraint had to be met by models of autoepistemic theories. This had the effect that the specification of a model had to include a potentially infinite list of all the formulas of the form LP that were to be taken as true. The resulting lack of structure in the models made it extremely difficult to prove results concerning the models of particular autoepistemic theories.

However, it turns out that, for autoepistemic theories representing sets of beliefs satisfying certain stability conditions, we can define models that have much more structure. The conditions of interest are that (1) the set of beliefs is closed under ordinary logical consequence, (2) whenever a formula P is believed, it is believed that P is believed, and (3) whenever a formula P is not believed, it is believed that P is not believed. We have been able to show that a set of beliefs satisfying these conditions can be characterized by a set of possible worlds such that a formula is believed if it is true in every world in the set, and a formula of the form LP is true in a particular world if P is true in every world in the set.

The important consequence of this demonstration is that such a set of beliefs can be characterized by a finite set of finite possible worlds whenever the number of atomic formulas in the language is finite. This in turn lets us define finite models under the same conditions, whereas, under our first definition, the models are finite only if the entire set of beliefs is finite.

With finite models, we can explore certain questions that are much harder to address with the infinite models of our original approach. For instance, consider what beliefs would be justified on the basis of the set of premises $\{\sim LP \rightarrow Q, \sim LQ \rightarrow P\}$. Informally speaking, these formulas say "If I don't believe P then Q is true" and "If I don't believe Q then P is true." Suppose these are an ideally rational agent's only premises. If he does not believe P, he can reflect on the fact that he does not believe P and he will conclude that Q is true. Conversely, if he does not believe Q, he can reflect on that and conclude that P is true. Thus it seems that he has grounds for believing P only if he does not believe Q and vice versa. So there are apparently two possible stable belief states that can be based on these premises. With the possible-world semantics for autoepistemic logic, we can demonstrate such conclusions rigorously by examining all the possible-world models of the premises. The details are presented in a recent paper [10], included as Appendix E.

2. A Weak Logic of Knowledge and Belief

Beginning with the work of Jaako Hintikka in the early 1960's [11], a number of attempts have been made to formulate and analyze varying conceptions of knowledge and belief by using the techniques of modal logic. In such research, the relevant notions are symbolized by one place modal or intensional operators on sentences. Various axioms governing these operators are then proposed. The important methodological conception is that one will be able to apply fairly standard techniques and results from the study of modal logic to the analysis of, and comparison between, such systems. Indeed, most

proposed systems have been exact analogues of one or another standard modal logic; that is, one simply replaces the modal operator for necessity with that for knowledge or belief. In the case of belief one must drop the analogue of the basic modal principle that, if it's necessary that P, then P. There are, after all, false beliefs.

Though we cannot reasonably idealize away false beliefs, any logic of knowledge and/or belief will have to embody some degree of idealization. Still it has seemed to many that the commitment to fairly standard modal systems has brought with it thoroughly inappropriate idealizations. Two distinct dimensions of idealization have been noted, and the locations of most proposed logics of knowledge and belief along these two dimensions have been criticized.

All standard modal logics or logics of necessity have been extensions of the system called K, which is the minimal modal logic. When conceived of as a basis for logics of knowledge and belief, this system yields the result that the subjects or agents in the intended domain of the theory know or believe all classical logical tautologies and, further, know or believe all the classical tautological consequences of anything they know or believe. With respect to the logic of necessity, these results are widely accepted. Surely all tautologies are necessarily true and, surely, if something is a logical consequence of a necessary truth, then it is itself a necessary truth. This has seemed to many to be a wildly inappropriate requirement on knowledge and/or belief. Unfortunately, committing oneself to working within modal logics weaker than K involves giving up some, perhaps much, of the power of analysis yielded by standard techniques in the theory of modal logics.

The other dimension of idealization has been that of "introspective" (or reflective) competence. How much are our subjects assumed to know or believe about their own knowledge and/or beliefs? Here, too, there has been a good deal of disagreement. With regard to knowledge, it has centered around the acceptability of the principle that, if one knows that P, one knows that one knows that P. (The

analogous principle in modal logic is that, if it is necessary that P, then it is necessary that it is necessary that P.) With regard to belief, a further locus of controversy has been the negative counterpart of the foregoing principle; namely that, if one doesn't believe that P, then one believes that one doesn't believe that P. The analogous principle about necessity is itself controversial.

Under this project, we have explored less drastic idealizations along the dimension of introspective competence. The considerations motivating commitment to the system K as a base are largely purely technical or tactical--the main point being simply a desire to separate problems that are separable in principle. With respect to knowledge, we suggest that one should begin, at least, with no more than the basic system K together with the principle that, if one knows that P, then P. In the case of belief, more drastic deviations from standard systems are proposed. In particular, a new axiom--called Y--is suggested. In one formulation, the axiom amounts to the following: if one believes that P, then one doesn't believe that one doesn't believe that P. This formulation brings out an essential feature of the proposed system: As an alternative to idealizing in such a way as to guarantee great scope to veridical introspection, the suggestion is to idealize in such a way as to guarantee against false introspective beliefs.

Considerations in favor of such an alternative idealization come from a number of sources; two, in particular, are the Paradox of the Preface and, most centrally, Moore's Paradox. The principle underlying the former is that we don't believe that all of our beliefs are true. Indeed, surely it's irrational for us to believe that we are in no way mistaken in our beliefs. Then we must reject the following principle: we believe that, if we believe that P, then P. (The analogous principle for knowledge is obviously correct.) Moore's paradox consists in this: It is odd or self-defeating for someone to assert both P and that he doesn't believe that P. (That is, any utterance of any instance of the sentence form "P; but I don't believe

that P." is, in some sense, self-denying.) The moral of Moore's paradox, at least with respect to the logic of belief, is that we do not believe of any one of our beliefs that we don't believe it. This is precisely the point of the axiom Y.

The full development of these ideas is presented in a paper by Israel [12], included as Appendix F. Israel characterizes the axiom Y, and the resulting system $K + Y$ in terms of the now standard model-theoretic techniques for modal logic. This yields both soundness and completeness results. He shows in what ways the formalization is weaker than standard logics of belief and sketches briefly some more general considerations about the appropriateness, given varying conceptions of the role of beliefs in action, of modal logics of knowledge and belief.

3. Plan Synthesis

Part of our work deals with techniques for automatic planning. Previous work in this vein has been highly experimental in nature, the standard methodology being to explore possible techniques by constructing working programs. Because of the emphasis on experimentation, very little has been done to analyze the techniques to determine why they work, when they are applicable, and whether it is possible to generalize them to solve larger classes of problems. Our work provides at least part of the missing analysis and introduces new techniques for plan synthesis.

We have approached the question of automatic planning from a rigorous, mathematical standpoint. Our methodology has been to develop a mathematical framework in which to study planning problems, to explore this framework for theorems that can be used to constrain the search for a solution, and then to construct planning techniques based on the theorems that were found. By following this methodology, it has been possible to develop techniques (a) that are capable of solving a much broader class of problems than had previously been considered, and (b) that are guaranteed to find a solution if one exists. Furthermore, it has been possible to unify many existing ideas in automatic planning, showing how these ideas arise from first principles.

The mathematical framework that has been developed is very much like that of first-order dynamic logic [13]. In this framework, the world may be in any one of a possibly infinite number of states. Performing an action causes the world to jump from one state to another. A planning problem in this framework consists of a description of the initial state, a description of the goal state, and a description of the allowable actions. The problem is to find a sequence of actions that is guaranteed to force the world into a state satisfying the goal description, given that the world may initially be in any one of the states satisfying the initial-state description. (State descriptions may be incomplete; that is, there may be more than one state satisfying a given description.)

Formally, a state description is a set of formulae in first-order logic, and a state is a first-order model. Actions are binary relations on states. For planning purposes, though, all that we need to know about an action are its preconditions and its regression operator. The preconditions of an action are a set of formulae defining the states in which the action may be performed. A regression operator for an action is a function mapping formulae to formulae such that the regression of a formula is the weakest condition that must be true before the action is performed in order for the formula to be true afterward. One of the contributions of our work is a language for describing the effects of an action and a way of computing regression operators from action descriptions in this language. The language is significant in that it combines the generality of the situation calculus [14] with the notational convenience of STRIPS [15]. This allows the frame problem of the situation calculus to be circumvented to the same extent that it can be done in STRIPS.

The planning techniques are based primarily on two observations. The first is that the world changes state only as the result of an action. Therefore, if a formula is false, it will become true only if an action makes it true. The second observation is that a plan must be finite since we would like our goals to be achieved at a

definite point in the future. Consequently, there will always be a last point in a plan when a formula becomes true if it becomes true at all. These observations lead us to the following theorem: a formula is true at a point P in a plan if and only if (1) the formula is true in the initial state and remains true until at least point P, or (2) there is an action prior to P that causes the formula to become true and the formula remains true thereafter until at least point P. This theorem tells us that, to construct a plan to achieve some goal, either we must introduce an action that makes the goal true or we must prevent the goal from becoming false if it is true initially. From this theorem it is possible to derive a planning technique. The details are presented in a paper by Pednault [16], included as Appendix G.

III PUBLICATIONS

Robert C. Moore, "Semantical Considerations on Nonmonotonic Logic," Artificial Intelligence, Vol. 25, No. 1, pp. 75-94 (January 1985).

Robert C. Moore, "A Formal Theory of Knowledge and Action," in Formal Theories of the Commonsense World, J. Hobbs and R. C. Moore, eds., pp. 319-358 (Ablex Publishing Corporation, Norwood, New Jersey, 1985).

Kurt Konolige, "Belief and Incompleteness," in Formal Theories of the Commonsense World, J. Hobbs and R. C. Moore, eds., pp. 359-403 (Ablex Publishing Corporation, Norwood, New Jersey, 1985).

Robert C. Moore, "Possible-World Semantics for Autoepistemic Logic," in the advance papers of the Non-Monotonic Reasoning Workshop sponsored by the American Association for Artificial Intelligence, New Paltz, New York, pp. 344-354 (October 17-19, 1984).

Other technical notes:

Kurt Konolige, "A Deduction Model of Belief and its Logics," Artificial Intelligence Center Technical Note 326, SRI International, Menlo Park, California (August 1984).

William Croft, "The Representation of Adverbs, Adjectives and Events in Logical Form," Artificial Intelligence Center Technical Note 344, SRI International, Menlo Park, California (December 1984).

David J. Israel, "A Weak Logic of Knowledge and Belief," Artificial Intelligence Center Technical Note 359, SRI International, Menlo Park, California (August 1985).

Edwin P. D. Pednault, "Preliminary Report on a Theory of Plan Synthesis," Artificial Intelligence Center Technical Note 358, SRI International, Menlo Park, California (August 1985).

In preparation:

Robert C. Moore, "Events, Situations, and Adverbs."

IV CONFERENCE PRESENTATIONS

Robert C. Moore, "Deductive Methods for Commonsense Reasoning," invited lecture, National Conference on Artificial Intelligence, Pittsburgh, Pennsylvania, August 18-20, 1982.

Robert C. Moore, "Semantical Considerations on Nonmonotonic Logic," Eighth International Joint Conference on Artificial Intelligence, Karlsruhe, West Germany, August 8-12, 1983.

Robert C. Moore, "Possible-World Semantics for Autoepistemic Logic," Non-Monotonic Reasoning Workshop, sponsored by the American Association for Artificial Intelligence, Mohonk Mountain House, New Paltz, New York, October 17-19, 1984.

V PERSONNEL

The research of this project has been carried out by Robert C. Moore (principal investigator), Kurt Konolige, and David Israel, with William Croft and Edwin Pednault as graduate research assistants. Supervision has been provided by Nils Nilsson, Stanley Rosenschein, and C. Raymond Perrault. Outside consultants have been Professor Patrick J. Hayes, University of Rochester; Professor Raymond Turner, University of Essex, U.K., and Professor C. Raymond Perrault, University of Toronto. (Professor Perrault joined SRI's permanent staff after having been a consultant on this project.)

Advanced degrees awarded:

Kurt Konolige, Ph.D., Department of Computer Science, Stanford University, June 1984, dissertation title: A Deduction Model of Belief and its Logics (partial support provided by this project).

REFERENCES

1. D. McDermott and J. Doyle, "Non-Monotonic Logic I," Artificial Intelligence, Vol. 13, Nos. 1, 2, pp. 41-72 (April 1980).
2. D. McDermott, "Nonmonotonic Logic II: Nonmonotonic Modal Theories," Journal of the Association for Computing Machinery, Vol. 29, No. 1, pp. 33-57 (January 1982).
3. R. C. Moore, "Semantical Considerations on Nonmonotonic Logic," Artificial Intelligence, Vol. 25, No. 1, pp. 75-94 (January 1985).
4. R. C. Moore, "Reasoning about Knowledge and Action," SRI Artificial Intelligence Center Technical Note 191, SRI International, Menlo Park, California (October 1980).
5. R. C. Moore, "A Formal Theory of Knowledge and Action," in Formal Theories of the Commonsense World, J. Hobbs and R. C. Moore, eds., pp. 319-358 (Ablex Publishing Corporation, Norwood, New Jersey, 1985).
6. D. Davidson, "The Logical Form of Action Sentences," in The Logic of Decision and Action, N. Rescher, ed., pp. 81-95 (University of Pittsburgh Press, Pittsburgh, Pennsylvania, 1967).
7. W. Croft, "The Representation of Adverbs, Adjectives and Events in Logical Form," Artificial Intelligence Center Technical Note 344, SRI International, Menlo Park, California (December 1984).
8. K. Konolige, A Deduction Model of Belief and its Logics, Ph.D. dissertation, Department of Computer Science, Stanford University (June 1984).
9. K. Konolige, "Belief and Incompleteness," in Formal Theories of the Commonsense World, J. Hobbs and R. C. Moore, eds., pp. 359-403 (Ablex Publishing Corporation, Norwood, New Jersey, 1985).
10. R. C. Moore, "Possible-World Semantics for Autoepistemic Logic," in the advance papers of the Non-Monotonic Reasoning Workshop sponsored by the American Association for Artificial Intelligence, New Paltz, New York, pp. 344-354 (October 17-19, 1984).
11. J. K. K. Hintikka, Knowledge and Belief (Cornell University Press, Ithaca, New York, 1962).
12. David J. Israel, "A Weak Logic of Knowledge and Belief," Artificial

Intelligence Center Technical Note 359, SRI International, Menlo Park, California (August 1985).

13. D. Harel, Lecture Notes in Computer Science, Volume 68: First-Order Dynamic Logic, New York: Springer-Verlag, 1979.
14. J. McCarthy and P. Hayes, "Some Philosophical Problems from the Standpoint of Artificial Intelligence," in Machine Intelligence 4, B. Meltzer and D. Michie, eds., pp. 463-502 (Edinburgh University Press, 1969).
15. R. E. Fikes and N. J. Nilsson, "SIRIPS: a New Approach to the Application of Theorem Proving to Problem Solving," Artificial Intelligence, Vol. 2, pp. 189-208, (1971).
16. E. P. D. Pednault, "Preliminary Report on a Theory of Plan Synthesis," Artificial Intelligence Center Technical Note 358, SRI International, Menlo Park, California (August 1985).

Appendix A

SEMANTICAL CONSIDERATION ON NONMONOTONIC LOGIC

SRI International



SEMANTICAL CONSIDERATIONS ON NONMONOTONIC LOGIC

Technical Note 284

June 1983

By: Robert C. Moore, Staff Scientist
Artificial Intelligence Center
Computer Science and Technology Division

SRI Project 4488

This is a revised and expanded version of a paper to appear in Proceedings of the Eighth International Joint Conference on Artificial Intelligence, Karlsruhe, West Germany, August 8-12, 1983.

The research reported herein was supported by the Air Force Office of Scientific Research under Contract No. F49620-82-K-0031. The views and conclusions expressed in this document are those of the author and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the Air Force Office of Scientific Research or the U.S. Government.

ABSTRACT

Commonsense reasoning is "nonmonotonic" in the sense that we often draw, on the basis of partial information, conclusions that we later retract when we are given more complete information. Some of the most interesting products of recent attempts to formalize nonmonotonic reasoning are the nonmonotonic logics of McDermott and Doyle [McDermott and Doyle, 1980; McDermott, 1982]. These logics, however, all have peculiarities that suggest they do not quite succeed in capturing the intuitions that prompted their development. In this paper we reconstruct nonmonotonic logic as a model of an ideally rational agent's reasoning about his own beliefs. For the resulting system, called autoepistemic logic, we define an intuitively based semantics for which we can show autoepistemic logic to be both sound and complete. We then compare autoepistemic logic with the approach of McDermott and Doyle, showing how it avoids the peculiarities of their nonmonotonic logic.

I INTRODUCTION

It has been generally acknowledged in recent years that one important feature of ordinary commonsense reasoning that standard logics fail to capture is its nonmonotonicity. An example frequently given to illustrate the point is the following. If we know that Tweety is a bird, we will normally assume, in the absence of evidence to the contrary, that Tweety can fly. If, however, we later learn that Tweety is a penguin, we will withdraw our prior assumption. If we try to model this in a formal system, we seem to have a situation in which a theorem P is derivable from a set of axioms S , but is not derivable from some set S' that is a superset of S . The set of theorems, therefore, does not increase monotonically with the set of axioms; hence this sort of reasoning is said to be "nonmonotonic." As Minsky [1974] has pointed out, standard logics are always monotonic, because their inference rules make every axiom permissive. That is, the inference rules are always of the form " P is a theorem if Q_1, \dots, Q_n are theorems," so that new axioms can only make more theorems derivable; they can never invalidate a previous theorem.

Recently there have been a number of attempts to formalize this type of nonmonotonic reasoning. The general idea is to allow axioms to be restrictive as well as permissive, by employing inference rules of the form " P is a theorem if Q_1, \dots, Q_n are not theorems." The inference that birds can fly is handled by having, in effect, a rule that says that, for any X , " X can fly" is a theorem if " X is a bird" is a theorem and " X cannot fly" is not a theorem. If all we are told about Tweety is that he is a bird, we will not be able to derive "Tweety cannot fly"; consequently, "Tweety can fly" will be inferable. If we are told that Tweety is a penguin and we already know that no penguin can fly, we will be able to derive the fact that Tweety cannot fly, and so the inference that Tweety can fly will be blocked.

One of the most interesting embodiments of this approach to nonmonotonic reasoning is McDermott and Doyle's "nonmonotonic logic" [McDermott and Doyle, 1980; McDermott, 1982]. McDermott and Doyle modify a standard first-order logic by introducing a sentential operator "M," whose informal interpretation is "is consistent." Nonmonotonic inferences about birds being able to fly would be sanctioned in their system by the axiom [McDermott, 1982, p. 33]

$$(ALL X)(BIRD(X) /\ M(CAN-FLY(X)) \rightarrow CAN-FLY(X)).$$

This formula can be read informally as "for all X, if X is a bird and it is consistent to assert that X can fly, then X can fly." McDermott and Doyle can then have a single general nonmonotonic inference rule, whose intuitive content is "MP is derivable if ~P is not derivable."

McDermott and Doyle's approach to nonmonotonic reasoning seems more interesting and ambitious than some other approaches in two respects. First, since the principles that lead to nonmonotonic inferences are explicitly represented in the logic, those very principles can be reasoned about. That is, if P is such a principle, we could start out believing $Q \rightarrow P$ or even $MP \rightarrow P$, and come to hold P by drawing inferences, either monotonic or nonmonotonic. So, if we use McDermott and Doyle's representation of the belief that birds can fly, we could also represent various inferences that would lead us to adopt that belief. Second, since they use only general inference rules, they are able to provide a formal semantic interpretation with soundness and completeness proofs for each of the logics they define. In formalisms that use content-specific nonmonotonic inference rules dealing with contingent aspects of the world (i.e., it might have been the case that birds could not fly), it is difficult to see how this could be done. The effect is that nonmonotonic inferences in McDermott and Doyle's logics are justified by the meaning of the premises of the inferences.

There are a number of problems with McDermott and Doyle's nonmonotonic logics, however. The first logic they define [McDermott and Doyle, 1980] gives such a weak notion of consistency that, as they

point out, MP is not inconsistent with $\sim P$. That is, it is possible for a theory to assert simultaneously that P is consistent with the theory and that P is false. McDermott subsequently [1982] tried basing nonmonotonic logics on the standard modal logics T, S4, and S5. He discovered, however, that the most plausible candidate for formalizing the notion of consistency that he wanted, nonmonotonic S5, collapses to ordinary S5 and is therefore monotonic. In the rest of this paper we develop an alternative formalization of nonmonotonic logic that shows why these problems arise in McDermott and Doyle's logics and how they can be avoided.

II NONMONOTONIC LOGIC AND AUTOEPISTEMIC REASONING

The first step in analyzing nonmonotonic logic is to determine what sort of nonmonotonic reasoning it is meant to model. After all, nonmonotonicity is a rather abstract syntactic property of an inference system, and there is no a priori reason to believe that all forms of nonmonotonic reasoning should have the same logical basis. In fact, McDermott and Doyle seem to confuse two quite distinct forms of nonmonotonic reasoning, which we will call default reasoning and autoepistemic reasoning. They talk as though their systems were intended to model the former, but they actually seem much better suited to modeling the latter.

By default reasoning we mean the drawing of plausible inferences from less-than-conclusive evidence in the absence of information to the contrary. The examples about birds being able to fly are of this type. If we know that Tweety is a bird, that gives us some evidence that Tweety can fly, but it is not conclusive. In the absence of information to the contrary, however, we are willing to go ahead and tentatively conclude that Tweety can fly. Now even before we do any detailed analysis of nonmonotonic logic, we can see that there will be problems in interpreting it as a model of default reasoning: In the formal semantics McDermott and Doyle provide for nonmonotonic logic, all the nonmonotonic inferences are valid. Default reasoning, however, is clearly not a form of valid inference.¹

Consider the belief that lies behind our willingness to infer that Tweety can fly from the fact that Tweety is a bird. It is probably something like most birds can fly, or almost all birds can fly, or a typical bird can fly. To model this kind of reasoning, in a theory whose only axioms are "Tweety is a bird" and "Most birds can fly," we ought to be able to infer (nonmonotonically) "Tweety can fly." Now if

this were a form of valid inference, we would be guaranteed that the conclusion is true if the premises are true. This is manifestly not the case. The premises of this inference give us a good reason to draw the conclusion, but not the ironclad guarantee that validity demands.

Now reconsider McDermott's formula that yields nonmonotonic inferences about birds being able to fly:

$$(ALL X)(BIRD(X) /\ M(CAN-FLY(X)) \rightarrow CAN-FLY(X))$$

McDermott suggests as a gloss of this formula "Most birds can fly," which would indicate that he thinks of the inferences it sanctions as default inferences. But if we read M as "is consistent" as McDermott and Doyle repeatedly tell us to do elsewhere, the formula actually says something quite different: "For all X, if X is a bird and it is consistent to assert that X can fly, then X can fly." Since the inference rule for M is intended to convey "MP is derivable if ~P is not derivable," the notion of consistency McDermott and Doyle have in mind seems to be that it is consistent to assert P if ~P is not derivable. McDermott's formula, then, says that the only birds that cannot fly are the ones that can be inferred not to fly. If we have a theory whose only axioms are this one and an assertion to the effect that Tweety is a bird, then the conclusion that Tweety can fly would be a valid inference. That is, if it is true that Tweety is a bird, and it is true that only birds inferred not to fly are in fact unable to fly, and Tweety is not inferred not to fly, then it must be true that Tweety can fly.

This type of reasoning is not a form of default reasoning at all; it rather seems to be more like reasoning about one's own knowledge or belief. Hence, we will refer to it as autoepistemic reasoning. Autoepistemic reasoning, while different from default reasoning, is an important form of commonsense reasoning in its own right. Consider my reason for believing that I do not have an older brother. It is surely not that one of my parents once casually remarked, "You know, you don't have any older brothers," nor have I pieced it together by carefully

sifting other evidence. I simply believe that if I did have an older brother I would know about it; therefore, since I don't know of any older brothers, I must not have any. This is quite different from a default inference based on the belief, say, that most MIT graduates are eldest sons, and that, since I am an MIT graduate, I am probably an eldest son.

Default reasoning and autoepistemic reasoning are both nonmonotonic, but for different reasons. Default reasoning is nonmonotonic because, to use a term from philosophy, it is defeasible: its conclusions are tentative, so, given better information, they may be withdrawn. Purely autoepistemic reasoning, however, is not defeasible. If you really believe that you already know all the instances of birds that cannot fly, you cannot consistently hold to that belief and at the same time accept new instances of birds that cannot fly.²

As Stalnaker [1980] has observed, autoepistemic reasoning is nonmonotonic because the meaning of an autoepistemic statement is context-sensitive; it depends on the theory in which the statement is embedded.³ If we have a theory whose only two axioms are

BIRD(TWEETY)
 (ALL X)(BIRD(X) /\ M(CAN-FLY(X)) -> CAN-FLY(X)),

then MP does not merely mean that P is consistent--it means that P is consistent with the nonmonotonic theory that contains only those two axioms. We would expect CAN-FLY(TWEETY) to be a theorem of this theory. If we change the theory by adding ~CAN-FLY(TWEETY) as an axiom, we then change the meaning of MP to be that P is consistent with the nonmonotonic theory that contains only the axioms

~CAN-FLY(TWEETY)
 BIRD(TWEETY)
 (ALL X)(BIRD(X) /\ M(CAN-FLY(X)) -> CAN-FLY(X)),

and we would not expect CAN-FLY(TWEETY) to be a theorem. The operator M changes its meaning with context just as do indexical words in natural language, such as "I," "here," and "now." The nonmonotonicity

associated with autoepistemic statements should therefore be no more puzzling than the fact that "I am hungry" can be true when uttered by a particular speaker at a particular time, but false when uttered by a different speaker at the same time or the same speaker at a different time. So we might say that, whereas default reasoning is nonmonotonic because it is defeasible, autoepistemic reasoning is nonmonotonic because it is indexical.

III THE FORMALIZATION OF AUTOEPISTEMIC LOGIC

Rather than try directly to analyze McDermott and Doyle's nonmonotonic logic as a model of autoepistemic reasoning, we will first define a logic that demonstrably does model certain aspects of autoepistemic reasoning and then compare nonmonotonic logic with that. We will call our logic, naturally enough, autoepistemic logic. The language will be much like McDermott and Doyle's, an ordinary logical language augmented by autoepistemic modal operators. McDermott and Doyle treat consistency as their fundamental notion, so they take M as the basic modal operator and define its dual L to be $\sim M$. Our logic, however, will be based on the notion of belief, so we will take L to mean "is believed," treat it as primitive, and define M as $\sim L$. In any case, this gives us the same notion of consistency as theirs: a formula is consistent if its negation is not believed. Since there are some problems with regard to the meaning of quantifying into the scope of an autoepistemic operator that are not relevant to the main point of this paper, we will limit our attention to propositional autoepistemic logic.

Autoepistemic logic is intended to model the beliefs of an agent reflecting upon his own beliefs. The primary objects of interest are sets of autoepistemic logic formulas that are interpreted as the total beliefs of such agents. We will call such a set of formulas an autoepistemic theory. The truth of an agent's beliefs, expressed as a propositional autoepistemic theory, will be determined by (1) which propositional constants are true in the external world and (2) which formulas the agent believes. A formula of the form LP will be true with respect to an agent if and only if P is in his set of beliefs. To formalize this, we define notions of interpretation and model as follows:

We proceed in two stages. First we define a propositional interpretation of an autoepistemic theory T to be an assignment of truth-values to the formulas of the language of T that is consistent with the usual truth recursion for propositional logic and with any arbitrary assignment of truth-values to propositional constants and formulas of the form LP . A propositional model of an autoepistemic theory T is a propositional interpretation of T in which all the formulas of T are true. The propositional interpretations and models of an autoepistemic theory are, therefore, precisely those we would get in ordinary propositional logic by treating all formulas of the form LP as propositional constants. We therefore inherit the soundness and completeness theorems of propositional logic; i.e., a formula P is true in all the propositional models of an autoepistemic theory T if and only if it is a tautological consequence of T (i.e., derivable from T by the usual rules of propositional logic).

Next we define an autoepistemic interpretation of an autoepistemic theory T to be a propositional interpretation of T in which, for every formula P , LP is true if and only if P is in T . It should be noted that the theory T itself completely determines the truth of any formula of the form LP in all the autoepistemic interpretations of T , independently of the truth assignment to the propositional constants. Hence, for every truth assignment to the propositional constants of T , there is exactly one corresponding autoepistemic interpretation of T . Finally, an autoepistemic model of T is an autoepistemic interpretation of T in which all the formulas of T are true. So the autoepistemic interpretations and models of T are just the propositional interpretations and models of T that conform to the intended meaning of the modal operator L .

This gives us a formal semantics for autoepistemic logic that matches its intuitive interpretation. Suppose that the beliefs of an agent situated in a particular world are characterized by the autoepistemic theory T . The world in question will provide an assignment of truth-values for the propositional constants of T , and any

formula of the form LP will be true relative to the agent just in case he believes P. In this way, the agent and the world in which he is situated directly determine an autoepistemic interpretation of T. That interpretation will be an autoepistemic model of T, just in case all the agent's beliefs are true in his world.

Given this semantics for autoepistemic logic, what do we want from a notion of inference for the logic? From an epistemological perspective, the problem of inference is the problem of what set of beliefs (theorems) an ideally rational agent would adopt on the basis of his initial premises (axioms). Since we are trying to model the beliefs of a rational agent, the beliefs should be sound with respect to the premises; we want a guarantee that the beliefs are true provided that the premises are true. Moreover, since we assume that the agent is ideally rational, the beliefs should be semantically complete; we want them to contain everything that the agent would be semantically justified in concluding from his beliefs and from the knowledge that they are his beliefs. An autoepistemic logic that meets these conditions can be viewed as a competence model of reflection upon one's own beliefs. Like competence models generally, it assumes unbounded resources of time and memory, and is therefore not a plausible model of any finite agent. It is, however, the model upon which the behavior of rational agents ought to converge as their time and memory resources increase.

Formally, we will say an autoepistemic theory T is sound with respect to an initial set of premises A if and only if every autoepistemic interpretation of T in which all the formulas of A are true is an autoepistemic model of T. This notion of soundness is the weakest condition that guarantees that all of the agent's beliefs are true whenever all his premises are true. Let I be the autoepistemic interpretation of T that is determined by what is true in the actual world (including what the agent actually believes). If all the formulas of T are true in every autoepistemic interpretation of T in which all the formulas of A are true, then all the formulas of T will be true in I

if all the formulas of A are true in I; hence, all the agent's beliefs will be true in the world if all the agent's premises are true in the world. However, if there is an autoepistemic interpretation of T in which all the formulas of A are true but some formulas of T are false, then it is possible that I is that interpretation, and that all the agent's premises will be true in the world, but some of his beliefs will not.

Our formal notion of completeness is that an autoepistemic theory T is semantically complete if and only if T contains every formula that is true in every autoepistemic model of T. If a formula P is true in every autoepistemic model of an agent's beliefs, then it must be true if all the agent's beliefs are true, and an ideally rational agent should be able to recognize that and infer P. On the other hand, if P is false in some autoepistemic model of the agent's beliefs, then that model, for all he can tell, might be the way the world actually is; he is therefore justified in not believing P.

The next problem is to give a syntactic characterization of the autoepistemic theories that satisfy these conditions. With a monotonic logic, the usual procedure is to define a collection of inference rules to apply to the axioms. For a nonmonotonic logic this is a nontrivial matter. Much of the technical ingenuity of McDermott and Doyle's systems lies simply in their formulation of a coherent notion of nonmonotonic derivability. The problem is that nonmonotonic inference rules do not yield a simple iterative notion of derivability the way monotonic inference rules do. We can view a monotonic inference process as applying the inference rules in all possible ways to the axioms, generating additional formulas to which the inference rules are applied in all possible ways, and so forth. Since monotonic inference rules are monotonic, once a formula has been generated at a given stage, it remains in the generated set of formulas at every subsequent stage. Thus the theorems of a theory in a monotonic system can be defined simply as all the formulas that are generated at any stage. The problem with attempting to follow this pattern with nonmonotonic inference rules

is that we cannot draw nonmonotonic inferences reliably at any particular stage, since something inferred at a later stage may invalidate them. Lacking such an iterative structure, nonmonotonic systems often use nonconstructive "fixed point" definitions, which do not directly yield algorithms for enumerating the "derivable" formulas, but do define sets of formulas that respect the intent of the nonmonotonic inference rules (e.g., in McDermott and Doyle's fixed points, MP is included whenever $\sim P$ is not included.)

For our logic, it is easiest to proceed by first specifying the closure conditions that we would expect the beliefs of an ideally rational agent to possess. Viewed informally, the beliefs should include whatever the agent could infer either by ordinary logic or by reflecting on what he believes. Stalnaker [1980] has put this formally by suggesting that a set of formulas T that represents the beliefs of an ideally rational agent should satisfy the following conditions:

1. If P_1, \dots, P_n are in T , and $P_1, \dots, P_n \vdash Q$, then Q is in T (where " \vdash " means ordinary tautological consequence).
2. If P is in T , then LP is in T .
3. If P is not in T , then $\sim LP$ is in T .

Stalnaker [1980, p. 6] describes the state of belief characterized by such a theory as stable "in the sense that no further conclusions could be drawn by an ideally rational agent in such a state." We will therefore describe the theories themselves as stable autoepistemic theories.

There are a number of interesting observations we can make about stable autoepistemic theories. First we note that, if a stable autoepistemic theory T is consistent, it will satisfy two more intuitively sound conditions:

4. If LP is in T , then P is in T .
5. If $\sim LP$ is in T , then P is not in T .

Condition 4 holds because, if LP were in T and P were not, $\neg LP$ would be in T (by Condition 3) and T would be inconsistent.⁴ Condition 5 holds because, if $\neg LP$ and P were both in T , LP would be in T (by Condition 2) and T would be inconsistent.

Conditions 2-5 imply that any consistent stable autoepistemic theory will be both sound and semantically complete with respect to formulas of the form LP and $\neg LP$: If T is such a theory, then LP will be in T if and only if P is in T , and $\neg LP$ will be in T if and only if P is not in T . Thus, all the propositional models of a stable autoepistemic theory are autoepistemic models. Stability implies a soundness result even stronger than this, however. We can show that the truth of any formula of a stable autoepistemic theory depends only on the truth of the formulas of the theory that contain no autoepistemic operators. (We will call these formulas "objective.")

Theorem 1. If T is a stable autoepistemic theory, then any autoepistemic interpretation of T that is a propositional model of the objective formulas of T is an autoepistemic model of T .

(The proofs of all theorems are given in the appendix.)

In other words, if all the objective formulas in an autoepistemic theory are true, then all the formulas in that theory are true. Given that the objective formulas of a stable autoepistemic theory determine whether the theory is true, it is not surprising that they also determine what all the formulas of the theory are.

Theorem 2. If two stable autoepistemic theories contain the same objective formulas, then they contain exactly the same formulas.⁵

Finally, with these characterization theorems, we can prove that the syntactic property of stability is equivalent the semantic property of completeness.

Theorem 3. An autoepistemic theory T is semantically complete if and only if T is stable.

By Theorem 3, we know that stability of an agent's beliefs guarantees that they are semantically complete, but stability alone does not tell us whether they are sound with respect to his initial premises. That is because the stability conditions say nothing about what an agent should not believe. They leave open the possibility of an agent's believing propositions that are not in any way grounded in his initial premises. What we need to add is a constraint specifying that the only propositions the agent believes are his initial premises and those required by the stability conditions. To satisfy the stability conditions and include a set of premises A, an autoepistemic theory T must include all the tautological consequences of $A \cup \{LP \mid P \text{ is in } T\} \cup \{\neg LP \mid P \text{ is not in } T\}$. Conversely, we will say that an autoepistemic theory T is grounded in a set of premises A if and only if every formula of T is included in the tautological consequences of $A \cup \{LP \mid P \text{ is in } T\} \cup \{\neg LP \mid P \text{ is not in } T\}$. The following theorem shows that this syntactic constraint on T and A captures the semantic notion of soundness.

Theorem 4. An autoepistemic theory T is sound with respect to an initial set of premises A if and only if T is grounded in A.

From Theorems 3 and 4, we can see that the possible sets of beliefs that an ideally rational agent might hold, given A as his premises, ought to be just the extensions of A that are grounded in A and stable. We will call these the stable expansions of A. Note that we say "sets", because there may be more than one stable expansion of a given set of premises. For example, consider $\{\neg LP \rightarrow Q, \neg LQ \rightarrow P\}$ as an initial set of premises.⁶ The first formula asserts that, if P is not believed, then Q is true; the second asserts that, if Q is not believed, then P is true. In any stable autoepistemic theory that includes these premises, if P is not in the theory, Q will be, and vice versa. But if the theory is grounded in these premises, if P is in the theory there will be no basis for including Q, and vice versa. Consequently, a stable expansion of $\{\neg LP \rightarrow Q, \neg LQ \rightarrow P\}$ will contain either P or Q, but not both.

It can also happen that there are no stable expansions of a given set of premises. Consider, for instance, $\{\neg LP \rightarrow P\}$.⁷ If T is a stable autoepistemic theory that contains $\neg LP \rightarrow P$, it must also contain P . If P were not in T , $\neg LP$ would have to be in the T , but then P would be in T --a contradiction. On the other hand, if P is in T , then T is not grounded in $\{\neg LP \rightarrow P\}$. Therefore no stable autoepistemic theory can be grounded in $\{\neg LP \rightarrow P\}$.

This seemingly strange behavior results from the indexicality of the autoepistemic operator L . Since L is interpreted relative to an entire set of beliefs, its interpretation will change with the various ways of completing a set of beliefs. In each acceptable completion of a set of beliefs, the interpretation of L will change to make that set stable and grounded in the premises. Sometimes, though, no matter how we try to form a complete a set of beliefs, the result never coincides with the interpretation of L in a way that gives us a stable set of beliefs grounded in the premises.

This raises the question of how to view autoepistemic logic as a logic. If we consider a set of premises A as axioms, what do we consider the theorems of A to be? If there is a unique stable expansion of A , it seems clear that we want this expansion to be the set of theorems of A . But what if there are several stable expansions of A --or none at all? If we take the point of view of the agent, we have to say that there can be alternative sets of theorems, or no set of theorems of A . This may be a strange property for a logic to possess, but, given our semantics, it is clear why this happens. An alternative (adopted by McDermott and Doyle with regard to their fixed points) is to take the theorems of A to be the intersection of the set of all formulas of the language with all the stable expansions of A . This yields the formulas that are in all stable expansions of A if there is more than one, and it makes the theory inconsistent if there is no stable expansion of A . This too is reasonable, but it has a different interpretation. It represents what an outside observer would know, given only knowledge of the agent's premises and that he is ideally rational.

IV ANALYSIS OF NONMONOTONIC LOGIC

Now we are in a position to provide an analysis of nonmonotonic logic that will explain its peculiarities in terms of autoepistemic logic. Briefly, our conclusions will be that the original nonmonotonic logic of McDermott and Doyle [1980] is simply too weak to capture the notions they wanted, and that McDermott's [1982] attempt to strengthen the logic does so in the wrong way.

McDermott and Doyle's first logic is very similar to our autoepistemic logic with one glaring exception; its specification includes nothing corresponding to our Condition 2 (if P is in T , then LP is in T). McDermott and Doyle define the nonmonotonic fixed points of a set of premises A , corresponding to our stable expansions of A . In the propositional case, their definition is equivalent to the following:

T is a fixed point of A just in case T is the set of tautological consequences of $A \cup \{\neg LP \mid P \text{ is not in } T\}$.

Our definition of a stable expansion of A , on the other hand, could be stated as

T is a stable expansion of A just in case T is the set of tautological consequences of $A \cup \{LP \mid P \text{ is in } T\} \cup \{\neg LP \mid P \text{ is not in } T\}$.

In nonmonotonic logic, $\{LP \mid P \text{ is in } T\}$ is missing from the "base" of the fixed points. This makes it possible for there to be nonmonotonic theories with fixed points that contain P but not LP . So, under an autoepistemic interpretation of L , McDermott and Doyle's agents are omniscient as to what they do not believe, but they may know nothing as to what they do believe.

This explains essentially all the peculiarities of McDermott and Doyle's original logic. For instance, they note [1980, p. 69] that MC does not follow from $M(C \wedge D)$. Changing the modality to L , this is

equivalent to saying that $\neg LP$ does not follow from $\neg L(P \vee Q)$. The problem is that, lacking the ability to infer LP from P , nonmonotonic logic permits interpretations of L that are more restricted than simple belief. Suppose we interpret L as "inferable in n or fewer steps" for some particular n . P might be inferable in exactly n steps, and $P \vee Q$ in $n+1$. According to this interpretation $\neg L(P \vee Q)$ would be true and $\neg LP$ would be false. Since this interpretation of L is consistent with McDermott and Doyle's definition of a fixed point, $\neg LP$ does not follow from $\neg L(P \vee Q)$. The other example of this kind noted by McDermott and Doyle is that $\{MC, \neg C\}$ has a consistent fixed point, which amounts to saying simultaneously that P is consistent with everything asserted and that P is false. But this set of premises is equivalent to $\{\neg LP, P\}$, which would have no consistent fixed points if LP were forced to be in every fixed point that contains P .

On the other hand, McDermott and Doyle consider it to be a problem that $\{MC \rightarrow D, \neg D\}$ has no consistent fixed point in their theory. Restated in terms of L , this set of premises is equivalent to $\{P \rightarrow L \neg P\}$. Since a stable autoepistemic theory containing these premises will also contain LQ , it must also contain Q to be consistent. (Otherwise it would contain $\neg LQ$.) But Q is not contained in any theory grounded in the premises $\{P \rightarrow LQ, P\}$; it is possible for $P \rightarrow LQ$ and P both to be true with respect to an agent while Q is false. So there is no consistent stable expansion of $\{P \rightarrow LQ, P\}$ in autoepistemic logic; hence, this set of premises cannot be the foundation of an appropriate set of beliefs for an ideally rational agent. Thus, our analysis justifies nonmonotonic logic in this case, contrary to the intuition of McDermott and Doyle.

McDermott and Doyle recognized the weakness of the original formulation of nonmonotonic logic, and McDermott [1982] has gone on to develop a group of theories that are stronger because they are based on modal rather than classical logic. McDermott's nonmonotonic modal theories alter the logic in two ways. First, the definition of fixed point is changed to be equivalent to

T is a fixed point of A just in case T is the set of modal consequences of $A \cup \{\sim LP \mid P \text{ is not in } T\}$,

where "modal consequence" means that $P \mid\sim LP$ is used as an additional inference rule. Second, McDermott considers only theories that include as premises the axioms of one of the standard modal logics "T," "S4," and "S5."

Merely changing the definition of fixed point brings McDermott's logic much closer to autoepistemic logic. In particular, adding $P \mid\sim LP$ as an inference rule means that all modal fixed points of A are stable expansions of A. However, adding $P \mid\sim LP$ as an inference rule, rather than adding $\{LP \mid P \text{ is in } T\}$ to the base of T, has as a consequence that not all stable expansions of A are modal fixed points of A. The difference is that, in autoepistemic logic, if P can be derived from LP, then both can be in a stable expansion of the premises, whereas in McDermott's logic there must be a derivation of P that does not rely on LP. Thus, although in autoepistemic logic there is a stable expansion of $\{LP \rightarrow P\}$ that includes P, in McDermott's logic there is no modal fixed point of $\{LP \rightarrow P\}$ that includes P. It is as if, in autoepistemic logic, one can acquire the belief that P and justify it later by the premise that, if P is believed, then it is true. In nonmonotonic logic, however, the justification of P has to precede belief in LP. This makes the interpretation of L in nonmonotonic modal logic more like "justified belief" than simple belief.

Since we have already shown that autoepistemic logic requires no specific axioms to capture a competence model of autoepistemic reasoning, we might wonder what purpose is served by McDermott's second modification of nonmonotonic logic, the addition of the axioms of various modal logics. The most plausible answer is that, besides behaving in accordance with the principles of autoepistemic logic, an ideally rational agent might well be expected to know what some of those principles are. For instance, the modal logic T has all instances of the schema $L(P \rightarrow Q) \rightarrow (LP \rightarrow LQ)$ as axioms. This says that the agent's beliefs are closed under modus ponens--which is true for an

ideally rational agent, so he might as well believe it. S4 adds the schema $LP \rightarrow LLP$, which means that, if the agent believes P, he believes that he believes it (Condition 2). S5 adds the schema $\neg LP \rightarrow L\neg LP$, which means that, if the agent does not believe P, he believes that he does not believe it (Condition 3). Since all these formulas are always true with respect to any ideally rational agent, it seems plausible to expect him to adopt them as premises. Thus, S5 seems to be the most plausible candidate of the nonmonotonic logics as a model of autoepistemic reasoning.

The problem is that all of these logics also contain the schema $LP \rightarrow P$, which means that, if the agent believes P, then P is true--but this is not generally true, even for ideally rational agents.⁸ It turns out that $LP \rightarrow P$ will always be contained in any stable autoepistemic theory (that is, ideally rational agents always believe that their beliefs are true), but making it a premise allows beliefs to be grounded that otherwise would not be. As a premise the schema $LP \rightarrow P$ can itself be justification for believing P, while as a "theorem" it must be derived from $\neg LP$, in which case P is not believed, or from P, in which case P must be independently justified, or from some other grounded formulas. In any case, as a premise schema, $LP \rightarrow P$ can sanction any belief whatsoever in autoepistemic logic. This is not generally true in modal nonmonotonic logic, as we have also seen, but it is true in nonmonotonic S5. The S5 axiom schema $\neg LP \rightarrow L\neg LP$ embodies enough of the model theory of autoepistemic logic to allow LP to be "self grounding": The schema $\neg LP \rightarrow L\neg LP$ is equivalent to the schema $\neg L\neg LP \rightarrow LP$, which allows LP to be justified by the fact that its negation is not believed. This inference is never in danger of being falsified, but, from this and $LP \rightarrow P$, we obtain an unwarranted justification for believing P.

The collapse of nonmonotonic S5 into monotonic S5 follows immediately. Since $LP \rightarrow P$ can be used to justify belief in any formula at all, there are no formulas that are absent from every fixed point of theories based on nonmonotonic S5. It follows that there are no formulas of the form $\neg LP$ that are contained in every fixed point of

theories based on nonmonotonic S5; hence there are no theorems of the form $\sim LP$ in any theory based on nonmonotonic S5. (Recall that the theorems are the intersection of all the fixed points.) Since these formulas are just the ones that would be produced by nonmonotonic inference, nonmonotonic S5 collapses to monotonic S5. In more informal terms, an agent who assumes that he is infallible is liable to believe anything, so an outside observer can conclude nothing about what he does not believe.

The real problem with nonmonotonic S5, then, is not the S5 schema; therefore McDermott's rather unmotivated suggestion to drop back to nonmonotonic S4 [1982, p. 45] is not the answer. The S5 schema merely makes explicit the consequences of adopting $LP \rightarrow P$ as a premise schema that are implicit in the logic's natural semantics. If we want to base nonmonotonic logic on a modal logic, the obvious solution is to drop back, not to S4, but to what Stalnaker [1980] calls "weak S5"--S5 without $LP \rightarrow P$. It is much better motivated and, moreover, has the advantage of actually being nonmonotonic.

In autoepistemic logic, however, even this much is unnecessary. Adopting any of the axioms of weak S5 as premises makes no difference to what can be derived. The key fact is the following theorem:

Theorem 5. If P is true in every autoepistemic interpretation of T , then T is grounded in $A \cup \{P\}$ if and only if T is grounded in A .

An immediate corollary of this result is that, if P is true in every autoepistemic interpretation of T , then T is a stable expansion of $A \cup \{P\}$ if and only if T is a stable expansion of A .

The modal axiom schemata of weak S5,

$$\begin{aligned} L(P \rightarrow Q) &\rightarrow (LP \rightarrow LQ) \\ LP &\rightarrow LLP \\ \sim LP &\rightarrow L\sim LP, \end{aligned}$$

simply state Conditions 1-3, so all their instances are true in every autoepistemic interpretation of any stable autoepistemic theory. The

nonmodal axioms of weak S5 are just the tautologies of propositional logic, so they are true in every interpretation (autoepistemic or otherwise) of any autoepistemic theory (stable or otherwise). It immediately follows by Theorem 5, therefore, that a set of premises containing any of the axioms of weak S5 will have exactly the same stable expansions as the corresponding set of premises without any weak-S5 axioms.

V CONCLUSION

McDermott and Doyle recognized that their original nonmonotonic logic was too weak; when McDermott tried to strengthen it, however, he misdiagnosed the problem. Because he was thinking of nonmonotonic logic as a logic of provability rather than belief, he apparently thought the problem was the lack of any connection between provability and truth. At one point he says "Even though M^{\sim}P (abbreviated LP) might plausibly be expected to mean 'P is provable,' there was not actually any relation between the truth values of P and LP," [1982, p. 34], and later he acknowledges the questionability of the schema $\text{LP} \rightarrow \text{P}$, but says that "it is difficult to visualize any other way of relating provability and truth," [1982, p. 35]. If one interprets nonmonotonic logic as a logic of belief, however, there is no reason to expect any connection between the truth of LP and the truth of P. And, as we have seen, the real problem with the original nonmonotonic logic was that the "if" half of the semantic definition of L--that LP is true if and only if P is believed--was not expressed in the logic.

NOTES

¹ In their informal exposition, McDermott and Doyle [1980, pp. 44-46] emphasize that their notion of nonmonotonic inference is not to be taken as a form of valid inference. If this is the case, their formal semantics cannot be regarded as the "real" semantics of their nonmonotonic logic. At best, it would provide the conditions that would have to hold for the inferences to be valid, but this leaves unanswered the question of what formulas of nonmonotonic logic actually mean.

² Of course, autoepistemic reasoning can be combined with default reasoning; we might believe that we know about most of the birds that cannot fly. This could lead to defeasible autoepistemic inferences, but their defeasibility would be the result of their also being default inferences.

³ Stalnaker's note, which to my knowledge remains unpublished, grew out of his comments as a respondent to McDermott at a Conference on Artificial Intelligence and Philosophy, held in March 1980 at the Center for Advanced Study in the Behavioral Sciences. N.B., the term "autoepistemic reasoning" is ours, not his.

⁴ Condition 4 will, of course, also be satisfied by an inconsistent stable autoepistemic theory, since such a theory would include all formulas of autoepistemic logic.

⁵ This theorem implies that our autoepistemic logic does not contain any "nongrounded" self-referential formulas, such as one finds in what are usually called "syntactical" treatments of belief. If, instead of a belief operator, we had a belief predicate, Bel, there might be a term p that denotes the formula $\text{Bel}(p)$. Whether $\text{Bel}(p)$ is believed or not is clearly independent of any objective beliefs. The lack of such formulas constitutes a characteristic difference between sentence-operator and predicate treatments of propositional attitudes and modalities.

⁶ McDermott and Doyle [1980, p. 51] present this example as $\{MC \rightarrow \neg D, MD \rightarrow \neg C\}$.

⁷ McDermott and Doyle [1980, p. 51] present this example as $\{MC \rightarrow \neg C\}$.

⁸ $LP \rightarrow P$ would be an appropriate axiom schema if the interpretation of LP were "P is known" rather than "P is believed," but that notion is not nonmonotonic. An agent cannot, in general, know when he does not know P, because he might believe P --leading him to believe that he knows P --while P is in fact false. Since agents are unable to reflect directly on what they do not know (only on what they do not believe), an autoepistemic logic of knowledge would not be a nonmonotonic logic; rather, the appropriate logic would seem to be monotonic S5.

REFERENCES

- McDermott, D. and J. Doyle [1980] "Non-Monotonic Logic I," Artificial Intelligence, Vol. 13, Nos. 1, 2, pp. 41-72 (April 1980).
- McDermott, D. [1982] "Nonmonotonic Logic II: Nonmonotonic Modal Theories," Journal of the Association for Computing Machinery, Vol. 29, No. 1, pp. 33-57 (January 1982).
- Minsky, M. [1974] "A Framework for Representing Knowledge," MIT Artificial Intelligence Laboratory, AIM-306, Massachusetts Institute of Technology, Cambridge, Massachusetts (June 1974).
- Stalnaker, R. [1980] "A Note on Non-monotonic Modal Logic," Dept. of Philosophy, Cornell University, unpublished manuscript.

APPENDIX: PROOFS OF THEOREMS

Theorem 1. If T is a stable autoepistemic theory, then any autoepistemic interpretation of T that is a propositional model of the objective formulas of T is an autoepistemic model of T .

Proof. Suppose that T is a stable autoepistemic theory and I is an autoepistemic interpretation of T that is a propositional model of the objective formulas of T . All the objective formulas of T are true in I . T must be consistent because an inconsistent stable autoepistemic theory would contain all formulas of the language, which would include many objective formulas that are not true in I . Let P be an arbitrary formula in T . Since stable autoepistemic theories are closed under tautological consequence, T must also contain a set of formulas P_1, \dots, P_k that taken together entail P , where, for each i between 1 and k , there exist n and m such that P_i is of the form

$$P_{i,1} \vee LP_{i,2} \vee \dots \vee LP_{i,n} \vee \sim LP_{i,n+1} \vee \dots \vee \sim LP_{i,m}$$

and $P_{i,1}$ is an objective formula. (Any formula is interderivable with a set of such formulas by propositional logic alone.) There are two cases to be considered:

(1) Suppose at least one of $LP_{i,2}, \dots, LP_{i,n}, \sim LP_{i,n+1}, \dots, \sim LP_{i,m}$ is in T . By Conditions 4 and 5, we know that, if any such formula is in T , it must be true in I , since T is consistent and I is an autoepistemic interpretation of T . But, since each of these formulas entails P_i , it follows that P_i is also true in I .

(2) Suppose the first case does not hold. Conditions 2 and 3 guarantee that in every stable autoepistemic theory, for every formula P , either LP or $\sim LP$ will be in the theory. Hence, if T does not contain any of $LP_{i,2}, \dots, LP_{i,n}, \sim LP_{i,n+1}, \dots, \sim LP_{i,m}$, it must contain all of $\sim LP_{i,2}, \dots, \sim LP_{i,n}, LP_{i,n+1}, \dots, LP_{i,m}$. But $P_{i,1}$ is a tautological consequence of P_i and these formulas (imagine repeated applications of

the resolution principle); so $P_{i,1}$ must be in T . But $P_{i,1}$ is objective, and so, by hypothesis, must be true in I . Since $P_{i,1}$ entails P_i , it must be the case that P_i is true in I .

In either case, P_i will be true in I . All the P_i taken together entail P , so P must also be true in I . Since P was chosen arbitrarily, every formula of T must be true in I ; hence I is an autoepistemic model of T .

Theorem 2. If two stable autoepistemic theories contain the same objective formulas, then they contain exactly the same formulas.

Proof. Suppose that T_1 and T_2 contain the same objective formulas and T_1 contains P . We prove by induction on the depth of nesting of autoepistemic operators in P (the "L-depth" of P) that T_2 also contains P . If the L-depth of P is 0, the theorem is trivially true, since P will be an objective formula. Now suppose that P has an L-depth of d greater than 0, and that, if two stable autoepistemic theories contain the same objective formulas, then they contain exactly the same formulas whose L-depth is less than d .

Since stable autoepistemic theories are closed under tautological consequence, T_1 must also contain a set of formulas P_1, \dots, P_k that are interderivable with P by propositional logic, where, for each i between 1 and k , there exist n and m such that P_i is of the form

$$P_{i,1} \vee LP_{i,2} \vee \dots \vee LP_{i,n} \vee \sim LP_{i,n+1} \vee \dots \vee \sim LP_{i,m}$$

and $P_{i,1}$ is an objective formula. Note that, since propositional logic will treat all the formulas of the form $LP_{i,j}$ as propositional constants, it is impossible to increase the L-depth of a formula by propositional inference, so each of these formulas will have an L-depth of not more than d .

We can also assume that T_1 and T_2 are consistent. If one of these theories were inconsistent, it would contain all formulas of the language. Since, by hypothesis, the two theories contain the same

objective formulas, the other theory would contain all the objective formulas of the language and, since these formulas are inconsistent, it would also contain all the formulas of the language. For each P_i , there are three cases to be considered:

(1) T_1 contains $LP_{i,j}$ for some j between 2 and n . Since T_1 is consistent, by Condition 4 it must also contain $P_{i,j}$. Since the L-depth of $P_{i,j}$ is one less than that of $LP_{i,j}$, it must be less than d ; so, by hypothesis, T_2 must contain $P_{i,j}$ and, by Condition 2, it must contain $LP_{i,j}$. But P_i is a tautological consequence of $LP_{i,j}$, so T_2 contains P_i .

(2) T_1 contains $\sim LP_{i,j}$ for some j between $n+1$ and m . Since T_1 is consistent, by Condition 5 it must not contain $P_{i,j}$. Since the L-depth of $P_{i,j}$ is one less than that of $\sim LP_{i,j}$, it must be less than d ; therefore, by hypothesis, T_2 must not contain $P_{i,j}$ and, by Condition 3, it must contain $\sim LP_{i,j}$. But P_i is a tautological consequence of $\sim LP_{i,j}$, so T_2 contains P_i .

(3) Suppose neither of the first two cases holds. Conditions 2 and 3 guarantee that in every stable autoepistemic theory, for every formula P , either LP or $\sim LP$ will be in the theory. Hence, if T_1 does not contain any of $LP_{i,2}, \dots, LP_{i,n}, \sim LP_{i,n+1}, \dots, \sim LP_{i,m}$, it must contain all of $\sim LP_{i,2}, \dots, \sim LP_{i,n}, LP_{i,n+1}, \dots, LP_{i,m}$. But $P_{i,1}$ is a tautological consequence of P_i and these formulas; so $P_{i,1}$ must be in T_1 . $P_{i,1}$ is objective, however, so $P_{i,1}$ must also be in T_2 . Since P_i is a tautological consequence of $P_{i,1}$, T_2 contains P_i .

Thus, all of P_1, \dots, P_k are in T_2 . Since P is a tautological consequence of these formulas, P is also in T_2 . Since P was chosen arbitrarily, every formula in T_1 is also in T_2 . The same argument can be used to show that every formula in T_2 is also in T_1 , so T_1 and T_2 contain exactly the same formulas.

Theorem 3. An autoepistemic theory T is semantically complete if and only if T is stable.

Proof. "If" direction: we show that, if T is a stable autoepistemic theory, then T contains every formula that is true in every autoepistemic model of T . Let T be a stable autoepistemic theory and let P be an arbitrary formula that is not in T . We show that there is an autoepistemic model of T in which P is false.

We know from propositional logic that P is propositionally equivalent to (i.e., true in the same propositional models as) the conjunction of a set of formulas P_1, \dots, P_k , where, for each i between 1 and k , there exist n and m such that P_i is of the form

$$P_{i,1} \vee LP_{i,2} \vee \dots \vee LP_{i,n} \vee \sim LP_{i,n+1} \vee \dots \vee \sim LP_{i,m}$$

and $P_{i,1}$ is an objective formula. Since P will be a tautological consequence of P_1, \dots, P_k and T is stable, Condition 1 guarantees that, if P is not in T , at least one of P_1, \dots, P_k must not be in T . Let P_i be such a formula. P_i is a tautological consequence of each of its disjuncts, so none of them can be in T . We show that there is an autoepistemic model of T in which all of these disjuncts are false.

Since $P_{i,1}$ is not in T , it must not be a tautological consequence of the objective formulas of T . Given this and the fact that $P_{i,1}$ is objective, it follows from the completeness theorem for propositional logic that there must be a truth assignment to the propositional constants of T in which $P_{i,1}$ is false and all the objective formulas of T are true. But, we can extend this truth assignment (or any truth assignment to the propositional constants of T --see Section III) to an autoepistemic interpretation of T . Call this interpretation I and note that $P_{i,1}$ is false in I . I will be a propositional model of the objective formulas of T ; so, by Theorem 1, I is an autoepistemic model of T in which $P_{i,1}$ is false.

Now consider the other disjuncts of P_i . Note that Conditions 2 and 3 require that a stable theory contain all the formulas of the form LP or $\sim LP$ that are true in the autoepistemic interpretations of the theory. Since none of $LP_{i,2}, \dots, LP_{i,n}, \sim LP_{i,n+1}, \dots, \sim LP_{i,m}$ are in T , none of $LP_{i,2}, \dots, LP_{i,n}, \sim LP_{i,n+1}, \dots, \sim LP_{i,m}$ are true in any autoepistemic

interpretation of T . In particular, none of $LP_{i,2}, \dots, LP_{i,n}, \neg LP_{i,n+1}, \dots, \neg LP_{i,m}$ are true in I . Therefore, I is an autoepistemic model of T in which, since all of the disjuncts of P_i are false, P_i itself is false. But P is propositionally equivalent to a conjunction that includes P_i , so I is an autoepistemic model of T in which P is false.

"Only if" direction: we show that, if T is semantically complete, then T is stable. Suppose T is semantically complete. For any formula P , if P is true in every autoepistemic model of T , then P is in T . Let I be an arbitrary autoepistemic model of T . If we can show that some formula P is true in I , P must be true in every autoepistemic model of T (since I is arbitrarily chosen) and, thus, P must be in T . We now show that T satisfies Conditions 1-3.

(1) Suppose P_1, \dots, P_n are in T and $P_1, \dots, P_n \vdash Q$. Since I is a model of T , P_1, \dots, P_n will be true in I . Since P_1, \dots, P_n will be true in I and Q is a tautological consequence of P_1, \dots, P_n , Q will also be true in I . Therefore, Q will be in T . (2) Suppose P is in T . Since I is an autoepistemic model of T , LP will be true in I . Therefore, LP will be in T . (3) Suppose P is not in T . Since I is an autoepistemic model of T , $\neg LP$ will be true in I . Therefore, $\neg LP$ will be in T .

Conditions 1-3 are all satisfied, so T is stable.

Theorem 4. An autoepistemic theory T is sound with respect to an initial set of premises A if and only if T is grounded in A .

Proof. "If" direction: suppose T is grounded in A . Every formula of T is therefore included in the tautological consequences of $A \cup \{LP \mid P \text{ is in } T\} \cup \{\neg LP \mid P \text{ is not in } T\}$. We show that T is sound with respect to A --i.e., that every autoepistemic interpretation of T in which all the formulas of A are true is an autoepistemic model of T .

Let I be an autoepistemic interpretation of T in which all the formulas in A are true. We show that I is an autoepistemic model of T . If P is in A , then, trivially, P is true in I . If P is of the form LQ

and Q is in T , or if P is of the form $\neg LQ$ and Q is not in T , then P is true in I because I is an autoepistemic interpretation of T . We have now shown that all the formulas in $A \cup \{LP \mid P \text{ is in } T\} \cup \{\neg LP \mid P \text{ is not in } T\}$ are true in I , so all their tautological consequences are true in I . But all the formulas of T are included in this set, so I is an autoepistemic model of T . Since I was an arbitrarily chosen autoepistemic interpretation of T in which all the formulas of A are true, every autoepistemic interpretation of T in which all the formulas of A are true is an autoepistemic model of T .

"Only if" direction: suppose T is sound with respect to A . Every autoepistemic interpretation of T in which all the formulas of A are true is therefore an autoepistemic model of T . We show that T is grounded in A --i.e., every formula of T is a tautological consequence of $A \cup \{LP \mid P \text{ is in } T\} \cup \{\neg LP \mid P \text{ is not in } T\}$.

Let $A' = A \cup \{LP \mid P \text{ is in } T\} \cup \{\neg LP \mid P \text{ is not in } T\}$. Note that, for all P , if P is in T , LP will be in A' , so LP will be true in every propositional model of A' ; however, if P is not in T , $\neg LP$ will be in A' and LP will not be true in any propositional model of A' . Therefore, in every propositional model of A' , LP is true if and only if P is in T , so every propositional model of A' is an autoepistemic interpretation of T . Since every autoepistemic interpretation of T in which all the formulas of A are true is an autoepistemic model of T , every propositional model of A' is an autoepistemic model of T . Since every formula in T is true in every autoepistemic model of T , every formula in T is true in every propositional model of A' . By the completeness theorem for propositional logic, every formula of T is therefore a tautological consequence of A' . Hence T is grounded in A .

Theorem 5. If P is true in every autoepistemic interpretation of T , then T is grounded in $A \cup \{P\}$ if and only if T is grounded in A .

Proof. Suppose that P is true in every autoepistemic interpretation of T . For any set of premises A , the set of autoepistemic interpretations of T in which every formula of $A \cup \{P\}$ is true is therefore the same as

the set of autoepistemic interpretations of T in which every formula of A is true. Thus, every autoepistemic interpretation of T in which every formula of $A \cup \{P\}$ is true is an autoepistemic model of T if and only if every autoepistemic interpretation of T in which every formula of A is true is an autoepistemic model of T . Hence, T is sound with respect to $A \cup \{P\}$ if and only if T is sound with respect to A . By Theorem 4, therefore, T is grounded in $A \cup \{P\}$ if and only if T is grounded in A .

Appendix B

A FORMAL THEORY OF KNOWLEDGE AND ACTION

SRI International



A FORMAL THEORY OF KNOWLEDGE AND ACTION

Technical Note 320

February 1984

By: Robert C. Moore, Staff Scientist
Artificial Intelligence Center
Computer Science and Technology Division

SRI Project 4488
SRI IR&D Project 500KZ

To appear in Formal Theories of the Commonsense World, J. R. Hobbs and R. C. Moore, eds., (Ablex Publishing Corp., Norwood, New Jersey, 1984).

The research reported herein was supported in part by the Air Force Office of Scientific Research under Contract No. F49620-82-K-0031. The views and conclusions expressed in this document are those of the author and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the Air Force Office of Scientific Research or the U.S. Government. This research was also made possible in part by a gift from the System Development Foundation as part of a coordinated research effort with the Center for the Study of Language and Information, Stanford University.

333 Ravenswood Ave. • Menlo Park, CA 94025
• 15 326-6200 • TWX 910-373-2046 • Telex 300486

ABSTRACT

Most work on planning and problem solving within the field of artificial intelligence assumes that the agent has complete knowledge of all relevant aspects of the problem domain and problem situation. In the real world, however, planning and acting must frequently be performed without complete knowledge. This imposes two additional burdens on an intelligent agent trying to act effectively. First, when the agent entertains a plan for achieving some goal, he must consider not only whether the physical prerequisites of the plan have been satisfied, but also whether he has all the information necessary to carry out the plan. Second, he must be able to reason about what he can do to obtain necessary information that he lacks. In this paper, we present a theory of action in which these problems are taken into account, showing how to formalize both the knowledge prerequisites of action and the effects of action on knowledge.

CONTENTS

ABSTRACT	ii
I THE INTERPLAY OF KNOWLEDGE AND ACTION	1
II FORMAL THEORIES OF KNOWLEDGE	6
A. A Modal Logic of Knowledge	6
B. A Possible-World Analysis of Knowledge	11
C. Knowledge, Equality, and Quantification	20
III FORMALIZING THE POSSIBLE-WORLD ANALYSIS OF KNOWLEDGE	29
A. Object Language and Metalanguage	29
B. A First-Order Theory of Knowledge	37
IV A POSSIBLE-WORLD ANALYSIS OF ACTION	42
V AN INTEGRATED THEORY OF KNOWLEDGE AND ACTION	54
A. The Dependence of Action on Knowledge	54
B. The Effects of Action on Knowledge	65
REFERENCES	83

I THE INTERPLAY OF KNOWLEDGE AND ACTION

Planning sequences of actions and reasoning about their effects is one of the most thoroughly studied areas within artificial intelligence (AI). Relatively little attention has been paid, however, to the important role that an agent's knowledge plays in planning and acting to achieve a goal. Virtually all AI planning systems are designed to operate with complete knowledge of all relevant aspects of the problem domain and problem situation. Often any statement that cannot be inferred to be true is assumed to be false. In the real world, however, planning and acting must frequently be performed without complete knowledge of the situation.

This imposes two additional burdens on an intelligent agent trying to act effectively. First, when the agent entertains a plan for achieving some goal, he must consider not only whether the physical prerequisites of the plan have been satisfied, but also whether he has all the information necessary to carry out the plan. Second, he must be able to reason about what he can do to obtain necessary information that he lacks. AI planning systems are usually based on the assumption that, if there is an action an agent is physically able to perform, and carrying out that action would result in the achievement of a goal P , then the agent can achieve P . With goals such as opening a safe,

however, there are actions that any human agent of normal abilities is physically capable of performing that would result in achievement of the goal (in this case, dialing the combination of the safe), but it would be highly misleading to claim that an agent could open a safe simply by dialing the combination unless he actually knew that combination. On the other hand, if the agent had a piece of paper on which the combination of the safe was written, he could open the safe by reading what was on the piece of paper and then dialing the combination, even if he did not know it previously.

In this paper, we will describe a formal theory of knowledge and action that is based on a general understanding of the relationship between the two.¹ The question of generality is somewhat problematical, since different actions obviously have different prerequisites and results that involve knowledge. What we will try to do is to set up a formalism in which very general conclusions can be drawn, once a certain minimum of information has been provided concerning the relation between specific actions and the knowledge of agents.

To see what this amounts to, consider the notion of a test. The essence of a test is that it is an action with a directly observable result that depends conditionally on an unobservable precondition. In the use of litmus paper to test the pH of a solution, the observable result is whether the paper has turned red or blue, and the unobservable precondition is whether the solution is acid or alkaline. What makes

such a test useful for acquiring knowledge is that the agent can infer whether the solution is acid or alkaline on the basis of his knowledge of the behavior of litmus paper and the observed color of the paper. When one is performing a test, it is this inferred knowledge, rather than what is directly observed, that is of primary interest.

If we tried to formalize the results of such a test by making simple assertions about what the agent knows subsequent to the action, we would have to include the result that the agent knows whether the solution is acid or alkaline as a separate assertion from the result that he knows the color of the paper. If we did this, however, we would completely miss the point that knowledge of the pH of the solution is inferred from other knowledge, rather than being a direct observation. In effect, we would be stipulating what actions can be used as tests, rather than creating a formalism within which we can infer what actions can be used as tests.

If we want a formal theory of how an agent's state of knowledge is changed by his performing a test, we have to represent and be able to draw inferences from the agent's having several independent pieces of information. Obviously, we have to represent that, after the test is performed, the agent knows the observable result. Furthermore, we have to represent the fact that he knows that the test has been performed. If he just walks into the room and sees the litmus paper on the table, he will know what color it is, but, unless he knows its recent history, he will not have gained any knowledge about the acidity of the solution.

We also need to represent the fact that the agent understands how the test works; that is, he knows how the observable result of the action depends on the unobservable precondition. Even if he sees the litmus paper put into the solution and then sees the paper change color, he still will not know whether the solution is acid or alkaline unless he knows how the color of the paper is related to the acidity of the solution. Finally, we must be able to infer that, if the agent knows (i) that the test took place, (ii) the observable result of the test, and (iii) how the observable result depends on the unobservable precondition, then he will know the unobservable precondition. Thus we must know enough about knowledge to tell us when an agent's knowing a certain collection of facts implies that he knows other facts as well.

From the preceding discussion, we can conclude that any formalism that enables us to draw inferences about tests at this level of detail must be able to represent the following types of assertions:

- (1) After A performs ACT, he knows whether Q is true.
- (2) After A performs ACT, he knows that he has just performed ACT.
- (3) A knows that Q will be true after he performs ACT if and only if P is true now.

Moreover, in order to infer what information an agent will gain as a result of performing a test, the formalism must embody, or be able to represent, general principles sufficient to conclude the following:

- (4) If (1), (2), and (3) are true, then, after performing ACT,
A will know whether P was true before he performed ACT.

It is important to emphasize that any work on these problems that is to be of real value must seek to elicit general principles. For instance, it would be possible to represent (1), (2), and (3) in an arbitrary, ad hoc manner and to add an axiom that explicitly states (4), thereby "capturing" the notion of a test. Such an approach, however, would simply restate the superficial observations put forth in this discussion. Our goal in this paper is to describe a formalism in which specific facts like (4) follow from the most basic principles of reasoning about knowledge and action.

II FORMAL THEORIES OF KNOWLEDGE

A. A Modal Logic of Knowledge

Since formalisms for reasoning about action have been studied extensively in AI, while formalisms for reasoning about knowledge have not, we will first address the problems of reasoning about knowledge. In Section III we will see that the formalism that we are led to as a solution to these problems turns out to be well suited to developing an integrated theory of knowledge and action.

The first step in devising a formalism for reasoning about knowledge is to decide what general properties of knowledge we want that formalism to capture. The properties of knowledge in which we will be most interested are those that are relevant to planning and acting. One such property is that anything that is known by someone must be true. If P is false, we would not want to say that anyone knows P. It might be that someone believes P or that someone believes he knows P, but it simply could not be the case that anyone knows P. This is, of course, a major difference between knowledge and belief. If we say that someone believes P, we are not committed to saying that P is either true or false, but if we say that someone knows P, we are committed to the truth of P. The reason that this distinction is important for planning and

acting is simply that, for an agent to achieve his goals, the beliefs on which he bases his actions must generally be true. After all, merely believing that performing a certain action will bring about a desired goal is not sufficient for being able to achieve the goal; the action must actually have the intended effect.

Another principle that turns out to be important for planning is that, if someone knows something, he knows that he knows it. This principle is often required for reasoning about plans consisting of several steps. Suppose an agent plans to use ACT₁ to achieve his goal, but, in order to perform ACT₁ he needs to know whether P is true and whether Q is true. Suppose, further, that he already knows that P is true and that he can find out whether Q is true by performing ACT₂. The agent needs to be able to reason that, after performing ACT₂, he will know whether P is true and whether Q is true. He knows that he will know whether Q is true because he understands the effects of ACT₂, but how does he know that he will know whether P is true? Presumably it works something like this: he knows that P is true, so he knows that he knows that P is true. If he knows how ACT₂ affects P, he knows that he will know whether P is true after he performs ACT₂. The key step in this argument is an instance of the principle that, if someone knows something, he knows that he knows it.

It might seem that we would also want to have the principle that, if someone does not know something, he knows that he does not know it-- but this turns out to be false. Suppose that A believes that P, but P is not true. Since P is false, A certainly does not know that P, but it is highly unlikely that he knows that he does not know, since he thinks that P is true.

Probably the most important fact about knowledge that we will want to capture is that agents can reason on the basis of their knowledge. All our examples depend on the assumption that, if an agent trying to solve a problem has all the relevant information, he will apply his knowledge to produce a solution. This creates a difficulty for us, however, since agents (at least human ones) are not, in fact, aware of all the logical consequences of their knowledge. The trouble is that we can never be sure which of the inferences an agent could draw, he actually will. The principle people normally use in reasoning about what other people know seems to be something like this: if we can infer that something is a consequence of what someone knows, then, lacking information to the contrary, we will assume that the other person can draw the same inference.

This suggests the adoption some sort of "default rule" (Reiter, 1980) for reasoning about what inferences agents actually draw, but, for the purposes of this study, we will make the simplifying assumption that agents actually do draw all logically valid inferences from their knowledge. We can regard this as the epistemological version of the

"frictionless case" in classical physics. For a more general framework in which weaker assumptions about the deductive abilities of agents can be expressed, see the work of Konolige (1984).

Finally, we will need to include the fact that these basic properties of knowledge are themselves common knowledge. By this we mean that everyone knows them, and everyone knows that everyone knows them, and everyone knows that everyone knows that everyone knows them, ad infinitum. This type of principle is obviously needed when reasoning about what someone knows about what someone else knows, but it is also important in planning, because an agent must be able to reason about what he will know at various times in the future. In such a case, his "future self" is analogous to another agent.

In his pioneering work on the logic of knowledge and belief, Hintikka (1962) presents a formalism that captures all these properties. We will define a formal logic based on Hintikka's ideas, but modified somewhat to be more compatible with the additional ideas of this paper. So, what follows is similar to the logic developed by Hintikka in spirit, but not in detail.

The language we will use initially is that of propositional logic, augmented by an operator KNOW and terms denoting agents. The formula KNOW(A,P) is interpreted to mean that the agent denoted by the term A knows the proposition expressed by the formula P. So, if JOHN denotes John and LIKES(BILL,MARY) means that Bill likes Mary,

KNOW(JOHN,LIKES(BILL,MARY)) means that John knows that Bill likes Mary. The axioms of the logic are inductively defined as all instances of the following schemata:

M1. P , such that P is an axiom of ordinary propositional logic

M2. $\text{KNOW}(A,P) \supset P$

M3. $\text{KNOW}(A,P) \supset \text{KNOW}(A,\text{KNOW}(A,P))$

M4. $\text{KNOW}(A,(P \supset Q)) \supset (\text{KNOW}(A,P) \supset \text{KNOW}(A,Q))$

closed under the principle that

M5. If P is an axiom, then $\text{KNOW}(A,P)$ is an axiom.

The closure of the axioms under the inference rule modus ponens (from $(P \supset Q)$ and P , infer Q) defines the theorems of the system. This system is very similar to those studied in modal logic. In fact, if A is held fixed, the resulting system is isomorphic to the modal logic S4 (Hughes and Cresswell, 1968). We will refer to this system as the modal logic of knowledge.

These axioms formalize in a straightforward way the principles for reasoning about knowledge that we have discussed. M2 says that anything that is known is true. M3 says that, if someone knows something, he knows that he knows it. M4 says that, if someone knows a formula P and a formula of the form $(P \supset Q)$, then he knows the corresponding formula Q . That is, everyone can (and does) apply modus ponens. M5 guarantees that the axioms are common knowledge. It first applies to M1-M4, which

says that everyone knows the basic facts about knowledge; however, since it also applies to its own output, we get axioms stating that everyone knows that everyone knows, etc. Since M5 applies to the axioms of propositional logic (M1), we can infer that everyone knows the facts they represent. Furthermore, because modus ponens is the only inference rule needed in propositional logic, the presence of M4 will enable us to infer that an agent knows any propositional consequence of his knowledge.

B. A Possible-World Analysis of Knowledge

We could try to use the modal logic of knowledge directly in a computational system for reasoning about knowledge and action, but, as we have argued elsewhere (Moore, 1980), all the obvious ways of doing this encounter difficulties. (Konolige's recent work (1984) suggests some new, more promising possibilities, but some important questions remain to be resolved.) There may well be solutions to these problems, but it turns out that they can be circumvented entirely by changing the language we use to describe what agents know. Instead of talking about the individual propositions that an agent knows, we will talk about what states of affairs are compatible with what he knows. In philosophy, these states of affairs are usually called "possible worlds," so we will adopt that term here as well.

This shift to describing knowledge in terms of possible worlds is based on a rich and elegant formal semantics for systems like our modal

logic of knowledge, which was developed by Hintikka (1962, 1971) in his work on knowledge and belief. The advantages of this approach are that it can be formalized within ordinary first-order classical logic in a way that permits the use of standard automatic-deduction techniques in a reasonably efficient manner² and that, moreover, it generalizes nicely to an integrated theory for describing the effects of actions on the agent's knowledge.

Possible-world semantics was first developed for the logic of necessity and possibility. From an intuitive standpoint, a possible world may be thought of as a set of circumstances that might have been true in the actual world. Kripke (1963) introduced the idea that a world should be regarded as possible, not absolutely, but only relative to other worlds. That is, the world W_1 might be a possible alternative to W_2 , but not to W_3 . The relation of one world's being a possible alternative to another is called the accessibility relation. Kripke then proved that the differences among some of the most important axiom systems for modal logic corresponded exactly to certain restrictions on the accessibility relation of the possible-world models of those systems. These results are reviewed in Kripke (1971). Concurrently with these developments, Hintikka (1962) published the first of his writings on the logic of knowledge and belief, which included a model theory that resembled Kripke's possible-world semantics. Hintikka's original semantics was done in terms of sets of sentences, which he

called model sets, rather than possible worlds. Later (Hintikka, 1971), however, he recast his semantics using Kripke's concepts, and it is that formulation we will use here.

Kripke's semantics for necessity and possibility can be converted into Hintikka's semantics for knowledge by changing the interpretation of the accessibility relation. To analyze statements of the form $\text{KNOW}(A, P)$, we will introduce a relation K , such that $K(A, W_1, W_2)$ means

that the possible world W_2 is compatible or consistent with what A knows

in the possible world W_1 . In other words, for all that A knows in W_1 ,

he might just as well be in W_2 . It is the set of worlds

$\{w_2 \mid K(A, W_1, w_2)\}$ that we will use to characterize what A knows in W_1 .

We will discuss A 's knowledge in W_1 in terms of this set, the set of

states of affairs that are consistent with his knowledge in W_1 , rather

than in terms of the set of propositions he knows. For the present, let

us assume that the first argument position of K admits the same set of

terms as the first argument position of KNOW . When we consider

quantifiers and equality, we will have to modify this assumption, but it

will do for now.

Introducing K is the key move in our analysis of statements about knowledge, so understanding what K means is particularly important. To

illustrate, suppose that in the actual world--call it W_0 --A knows that P, but does not know whether Q. If W_1 is a world where P is false, then W_1 is not compatible with what A knows in W_0 ; hence we would have $\neg K(A, W_1, W_0)$. Suppose that W_2 and W_3 are compatible with everything A knows, but that Q is true in W_2 and false in W_3 . Since A does not know whether Q is true, for all he knows, he might be in either W_2 or W_3 instead of W_0 . Hence, we would have both $K(A, W_2, W_0)$ and $K(A, W_3, W_0)$. This is depicted graphically in Figure 1.

Some of the properties of knowledge can be captured by putting constraints on the accessibility relation K. For instance, requiring that the actual world W_0 be compatible with what each knower knows in W_0 , i.e., $\forall a (K(a, W_0, W_0))$, is equivalent to saying that anything that is known is true. That is, if the actual world is compatible with what everyone [actually] knows, then no one has any false knowledge. This corresponds to the modal axiom M2.

The definition of K implies that, if A knows that P in W_0 , then P must be true in every world W_1 such that $K(A, W_1, W_0)$. To capture the fact that agents can reason with their knowledge, we will assume the

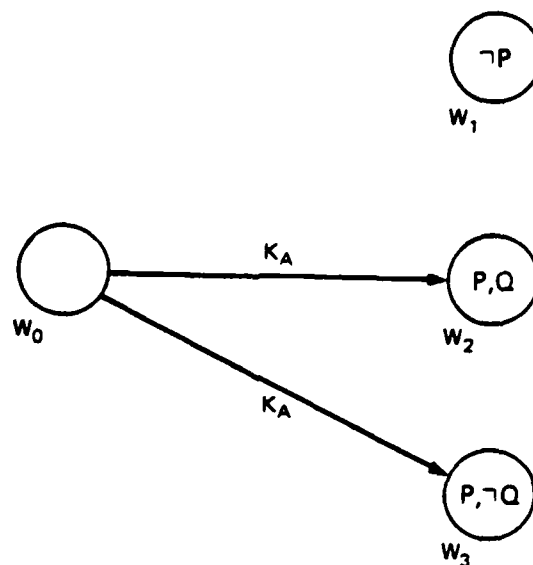


FIGURE 1 "A KNOWS THAT P"
"A DOESN'T KNOW WHETHER Q"

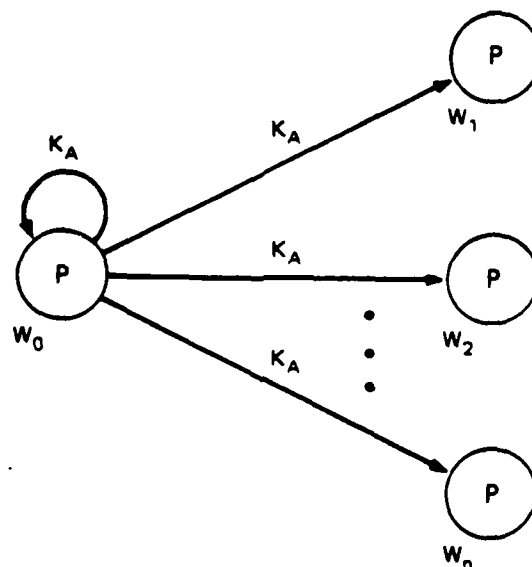


FIGURE 2 "P IS TRUE IN EVERY WORLD THAT IS COMPATIBLE WITH WHAT A KNOWS"

converse is also true. That is, we assume that, if P is true in every world W_1 such that $K(A, W_0, W_1)$, then A knows that P in W_0 . (See Figure 2.)

2.) This principle is the model-theoretic analogue of axiom M4 in the modal logic of knowledge. To see that this is so, suppose that A knows that P and that $(P \supset Q)$. Therefore, P and $(P \supset Q)$ are both true in every world that is compatible with what A knows. If this is the case, though, then Q must be true in every world that is compatible with what A knows. By our assumption, therefore, we conclude that A knows that Q .

Since this assumption, like M4, is equivalent to saying that an agent knows all the logical consequences of his knowledge, it should be interpreted only as a default rule. In a particular instance, the fact that P follows from A 's knowledge would be a justification for concluding that A knows P . However, we should be prepared to retract the conclusion that A knows P in the face of stronger evidence to the contrary.

With this assumption, we can get the effect of M3--the axiom stating that, if someone knows something, he knows that he knows it--by requiring that, for any W_1 and W_2 , if W_1 is compatible with what A knows in W_0 and W_2 is compatible with what A knows in W_1 , then W_2 is compatible with what A knows in W_0 . Formally expressed, this is

$$\forall a, w_1, w_2 (K(a, w_1, w_2) \supset (K(a, w_1, w_2) \supset K(a, w_1, w_2)))$$

By our previous assumption, the facts that A knows are those that are true in every world that is compatible with what A knows in the actual world. Furthermore, the facts that A knows that he knows are those that are true in every world that is compatible with what he knows in every world that is compatible with what he knows in the actual world. By the constraint we have just proposed, however, all these worlds must also be compatible with what A knows in the actual world (see Figure 3), so, if A knows that P, he knows that he knows that P.

Finally, we can get the effect of M5, the principle that the basic facts about knowledge are themselves common knowledge, by generalizing these constraints so that they hold not only for the actual world, but for all possible worlds. This follows from the fact that, if these constraints hold for all worlds, they hold for all worlds that are compatible with what anyone knows in the actual world; they also hold for all worlds that are compatible with what anyone knows in all worlds that are compatible with what anyone knows in the actual world, etc. Therefore, everyone knows the facts about knowledge that are represented by the constraints, and everyone knows that everyone knows, etc. Note that this generalization has the effect that the constraint corresponding to M2 becomes the requirement that, for a given knower, K is reflexive, while the constraint corresponding to M3 becomes the requirement that, for a given knower, K is transitive.

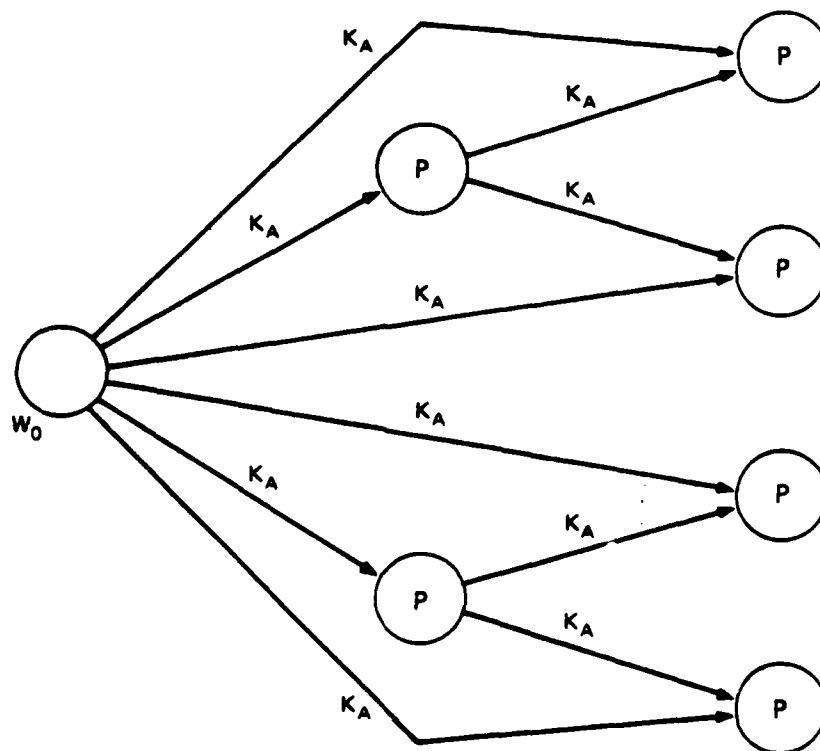


FIGURE 3 "IF A KNOWS THAT P, THEN HE KNOWS THAT HE KNOWS THAT P"

Analyzing knowledge in terms of possible worlds gives us a very nice treatment of knowledge about knowledge. Suppose A knows that B knows that P. Then, if the actual world is W_0 , in any world W_1 such that $K(A, W_0, W_1)$, B knows that P. We now continue the analysis relative to W_1 , so that, in any world W_2 such that $K(B, W_1, W_2)$, P is true. Putting both stages together, we obtain the analysis that, for any worlds W_1 and W_2 such that $K(A, W_0, W_1)$ and $K(B, W_1, W_2)$, P is true in W_2 . (See Figure 4.)

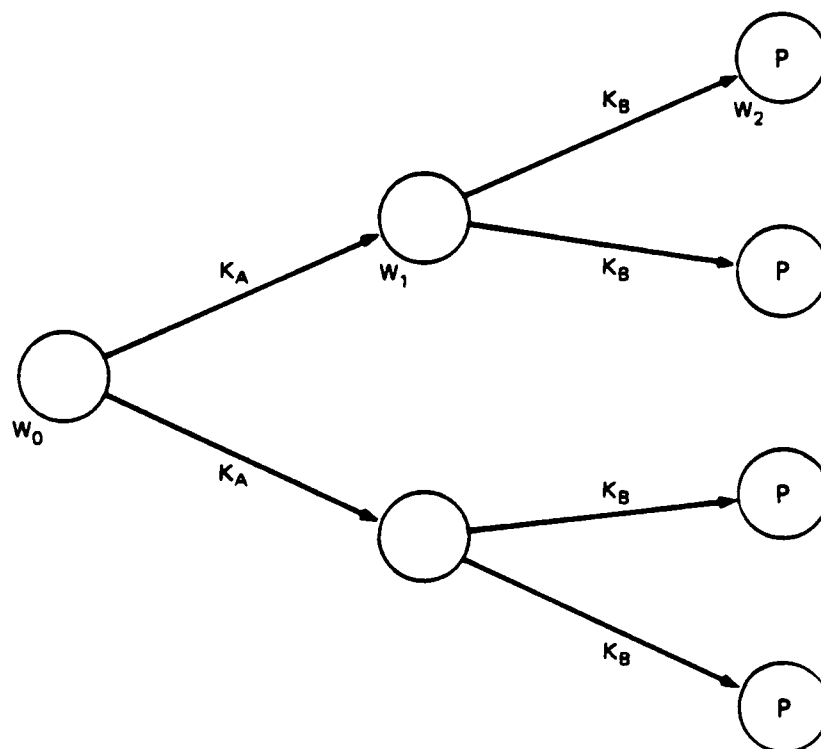


FIGURE 4 "A KNOWS THAT B KNOWS THAT P"

Given these constraints and assumptions, whenever we want to assert or deduce something that would be expressed in the modal logic of knowledge by $\text{KNOW}(A,P)$, we can instead assert or deduce that P is true in every world that is compatible with what A knows. We can express this in ordinary first-order logic, by treating possible worlds as individuals (in the logical sense), so that K is just an ordinary relation. We will therefore introduce an operator T such that $T(W,P)$ means that the formula P is true in the possible world W . If we let w_0 denote the actual world, we can convert the assertion $\text{KNOW}(A,P)$ into

$$\forall w_1 (K(A, w_0, w_1) \supset T(w_1, P))$$

It may seem that we have not made any real progress, since, although we have gotten rid of one nonstandard operator, KNOW, we have introduced another one, T. However, T has an important property that KNOW does not. Namely, T "distributes" over ordinary logical operators. In other words, $\neg P$ is true in W just in case P is not true in W , $(P \vee Q)$ is true in W just in case P is true in W or Q is true in W , and so on. We might say that T is extensional, relative to a possible world. This means that we can transform any formula so that T is applied only to atomic formulas. We can then turn T into an ordinary first-order relation by treating all the nonintensional atomic formulas as names of atomic propositions, or we can get rid of T by replacing the atomic formulas with predicates on possible worlds. This is no loss to the expressive power of the language, since, where we would have previously asserted P , we now simply assert $T(w_0, P)$ or $P(w_0)$ instead.

C. Knowledge, Equality, and Quantification

The formalization of knowledge presented so far is purely propositional; a number of additional problems arise when we attempt to extend the theory to handle equality and quantification. For instance, as Frege (1949) pointed out, attributions of knowledge and belief lead to violations of the principle of equality substitution. We are not entitled to infer $\text{KNOW}(A, P(C))$ from $B = C$ and $\text{KNOW}(A, P(B))$ because A might not know that the identity holds.

The possible-world analysis of knowledge provides a very neat solution to this problem, once we realize that a term can denote different objects in different possible worlds. For instance, if B is the expression "the number of planets" and C is "nine," then, although $B = C$ is true in the actual world, it would be false in a world in which there was a tenth planet. Thus, we will say that an equality statement such as $B = C$ is true in a possible world W just in case the denotation of the term B in W is the same as the denotation of the term C in W. This is a special case of the more general rule that a formula of the form $P(A_1, \dots, A_n)$ is true in W just in case the tuple consisting of the denotations in W of the terms A_1, \dots, A_n is in the extension in W of the relation expressed by P, provided that we fix the interpretation of $=$ in all possible worlds to be the identity relation.

Given this interpretation, the inference of $\text{KNOW}(A, P(C))$ from $B = C$ and $\text{KNOW}(A, P(B))$ will be blocked (as it should be). To infer $\text{KNOW}(A, P(C))$ from $\text{KNOW}(A, P(B))$ by identity substitution, we would have to know that B and C denote the same object in every world compatible with what A knows, but the truth of $B = C$ guarantees only that they denote the same object in the actual world. On the other hand, if $\text{KNOW}(A, P(B))$ and $\text{KNOW}(A, (B = C))$ are both true, then in all worlds that are compatible with what A knows, the denotation of B is in the extension of P and is the same as the denotation of C; hence, the denotation of C is in the extension of P. From this we can infer that $\text{KNOW}(A, P(C))$ is true.

The introduction of quantifiers also causes problems. To modify a famous example from Quine (1971), consider the sentence "Ralph knows that someone is a spy." This sentence has at least two interpretations. One is that Ralph knows that there is at least one person who is a spy, although he may have no idea who that person is. The other interpretation is that there is a particular person whom Ralph knows to be a spy. As Quine says (1971, p. 102), "The difference is vast; indeed, if Ralph is like most of us, [the first] is true and [the second] is false." This ambiguity was explained by Russell (1949) as a difference of scope. The idea is that indefinite noun phrases such as "someone" can be analyzed in context by paraphrasing sentences of the form $P(\text{"someone"})$ as "There exists a person x such that $P(x)$," or, more formally, $\exists x(\text{PERSON}(x) \wedge P(x))$. Russell goes on to point out that, in sentences of the form "A knows that someone is a P," the rule for eliminating "someone" can be applied to either the whole sentence or only the subordinate clause, "someone is a P." Applying this observation to "Ralph knows that someone is a spy," gives us the following two formal representations:

(1) $\text{KNOW}(\text{RALPH}, \exists x(\text{PERSON}(x) \wedge \text{SPY}(x)))$

(2) $\exists x(\text{PERSON}(x) \wedge \text{KNOW}(\text{RALPH}, \text{SPY}(x)))$

The most natural English paraphrases of these formulas are "Ralph knows that there is a person who is a spy," and "There is a person who

Ralph knows is a spy." These seem to correspond pretty well to the two interpretations of the original sentence. So, the ambiguity in the original sentence is mapped into an uncertainty as to the scope of the operator KNOW relative to the existential quantifier introduced by the indefinite description "someone."

Following a suggestion of Hintikka (1962), we can use a formula similar to (2) to express the fact that someone knows who or what something is. He points out that a sentence of the form "A knows who (or what) B is" intuitively seems to be equivalent to "there is someone (or something) that A knows to be B. But this can be represented formally as $\exists x(\text{KNOW}(A, (x = B)))$. To take a specific example, "John knows who the President is" can be paraphrased as "There is someone whom John knows to be the President," which can be represented by

(3) $\exists x(\text{KNOW}(\text{JOHN}, (x = \text{PRESIDENT})))$

In (1), KNOW may still be regarded as a purely propositional operator, although the proposition to which it is applied now has a quantifier in it. Put another way, KNOW still is used simply to express a relation between a knower and the proposition he knows. But (2) and (3) are not so simple. In these formulas there is a quantified variable that, although bound outside the scope of the operator KNOW, has an occurrence inside; this is sometimes called "quantifying in." Quantifying into knowledge and belief contexts is frequently held to pose serious problems of interpretation. Quine (1971), for instance,

holds that it is unintelligible, because we have not specified what proposition is known unless we say what description is used to fix the value of the quantified variable.

The possible-world analysis, however, provides us with a very natural interpretation of quantifying in. We keep the standard interpretation that $\exists x(P(x))$ is true just in case there is some value for x that satisfies P . If P is $\text{KNOW}(A, Q(x))$, then a value for x satisfies $P(x)$ just in case that value satisfies $Q(x)$ in every world that is compatible with what A knows. So (2) is satisfied if there is a particular person who is a spy in every world that is compatible with what A knows. That is, in every such world the same person is a spy. On the other hand, (1) is satisfied if, in every world compatible with what A knows, there is some person who is a spy, but it does not have to be the same one in each case.

Note that the difference between (1) and (2) has been transformed from a difference in the relative scopes of an existential quantifier and the operator KNOW to a difference in the relative scopes of an existential and a universal quantifier (the "every" in "every possible world compatible with..."). Recall from ordinary first-order logic that $\exists x(\forall y(P(x,y)))$ entails $\forall y(\exists x(P(x,y)))$, but not vice versa. The possible-world analysis, then, implies that we should be able to infer "Ralph knows that there is a spy," from "There is someone Ralph knows to be a spy," as indeed we can.

When we look at how this analysis applies to our representation for "knowing who," we get a particularly satisfying picture. We said that A knows who B is means that there is someone whom A knows to be B. If we analyze this, we conclude that there is a particular individual who is B in every world that is compatible with what A knows. Suppose this were not the case, and that, in some of the worlds compatible with what A knows, one person is B, whereas in the other worlds, some other person is B. In other words, for all that A knows, either of these two people might be B. But this is exactly what we mean when we say that A does not know who B is! Basically, the possible-world view gives us the very natural picture that A knows who B is if A has narrowed the possibilities for B down to a single individual.

Another consequence of this analysis worth noting is that, if A knows who B is and A knows who C is, we can conclude that A knows whether $B = C$. If A knows who B is and who C is, then B has the same denotation in all the worlds that are compatible with what A knows, and this is also true for C. Since, in all these worlds, B and C each have only one denotation, they either denote the same thing everywhere or denote different things everywhere. Thus, either $B = C$ is true in every world compatible with what A knows or $B \neq C$ is. From this we can infer that either A knows that B and C are the same individual or that they are not.

We now have a coherent account of quantifying in that is not framed in terms of knowing particular propositions. Still, in some cases

knowing a certain proposition counts as knowing something that would be expressed by quantifying in. For instance, the proposition that John knows that 321-1234 is Bill's telephone number might be represented as

(4) KNOW(JOHN, (321-1234 = PHONE-NUM(BILL))).

which does not involve quantifying in. We would want to be able to infer from this, however, that John knows what Bill's telephone number is, which would be represented as

(5) $\exists x$ (KNOW(JOHN, (x = PHONE-NUM(BILL)))).

It might seem that (5) can be derived from (4) simply by the logical principle of existential generalization, but that principle is not always valid in knowledge contexts. Suppose that (4) were not true, but that instead John simply knew that Mary and Bill had the same telephone number. We could represent this as

(6) KNOW(JOHN, (PHONE-NUM(MARY) = PHONE-NUM(BILL))).

It is clear that we would not want to infer from (6) that John knows what Bill's telephone number is--yet, if existential generalization were universally valid in knowledge contexts, this inference would go through.

It therefore seems that, in knowledge contexts, existential generalization can be applied to some referring expressions ("321-1234"), but not to others ("Mary's telephone number"). We will call the

expressions to which existential generalization can be applied standard identifiers, since they seem to be the ones an agent would use to identify an object for another agent. That is, "321-1234" is the kind of answer that would always be appropriate for telling someone what John's telephone number is, whereas "Mary's telephone number," as a

3

general rule, would not.

In terms of possible worlds, standard identifiers have a very straightforward interpretation. Standard identifiers are simply terms that have the same denotation in every possible world. Following Kripke (1972), we will call terms that have the same denotation in every possible world rigid designators. The conclusion that standard identifiers are rigid designators seems inescapable. If a particular expression can always be used by an agent to identify its referent for any other agent, then there must not be any possible circumstances under which it could refer to something else. Otherwise, the first agent could not be sure that the second was in a position to rule out those other possibilities.

The validity of existential generalization for standard identifiers follows immediately from their identification with rigid designators. The possible-world analysis of $\text{KNOW}(A, P(B))$ is that, in every world compatible with what A knows, the denotation of B in that world is in the extension of P in that world. Existential generalization fails in general because we are unable to conclude that there is any particular

individual that is in the extension of P in all the relevant worlds. If B is a rigid designator, however, the denotation of B is the same in every world. Consequently, it is the same in every world compatible with what A knows, and that denotation is an individual that is in the extension of P in all those worlds.

There are a few more observations to be made about standard identifiers and rigid designators. First, in describing standard identifiers we assumed that everyone knew what they referred to. Identifying them with rigid designators makes the stronger claim that what they refer to is common knowledge. That is, not only does everyone know what a particular standard identifier denotes, but everyone knows that everyone knows, etc. Second, although it is natural to think of any individual having a unique standard identifier, this is not required by our theory. What the theory does require is that, if there are two standard identifiers for the same individual, it should be common knowledge that they denote the same individual.

III FORMALIZING THE POSSIBLE-WORLD ANALYSIS OF KNOWLEDGE

A. Object Language and Metalanguage

As we indicated above, the analysis of knowledge in terms of possible worlds can be formalized completely within first-order logic by admitting possible worlds into the domain of quantification and making the extension of every expression depend on the possible world in which it is evaluated. For example, the possible-world analysis of "A knows who B is" would be as follows: There is some individual x such that, in every world w_1 that is compatible with what the agent who is A in the actual world knows in the actual world, x is B in w_1 . This means that in our formal theory we translate the formula of the modal logic of knowledge,

$$\exists x(\text{KNOW}(A, (x = B))),$$

into the first-order formula,

$$\exists x(\forall w_1 (K(A(w_0), w_0, w_1) \supset (x = B(w_1)))).$$

One convenient way of stating the translation rules precisely is to axiomatize them in our first-order theory of knowledge. This can be

done by introducing terms to denote formulas of the modal logic of knowledge (which we will henceforth call the object language) and axiomatizing a truth definition for those formulas in a first-order language that talks about possible worlds (the metalanguage). This has the advantage of letting us use either the modal language or the possible-world language--whichever is more convenient for a particular purpose--while rigorously defining the connection between the two.

The typical method of representing expressions of one formal language in another is to use string operations like concatenation or list operations like CONS in LISP, so that the conjunction of P and Q might be represented by something like CONS(P,CONS('A,CONS(Q,NIL))). which could be abbreviated LIST(P,'A,Q). This would be interpreted as a list whose elements are P followed by the conjunction symbol followed by Q. Thus, the metalanguage expression CONS(P,CONS('A,CONS(Q,NIL))) would denote the object language expression (P A Q). McCarthy (1962) has devised a much more elegant way to do the encoding, however. For purposes of semantic interpretation of the object language, which is what we want to do, the details of the syntax of that language are largely irrelevant. In particular, the only thing we need to know about the syntax of conjunctions is that there is some way of taking P and Q and producing the conjunction of P and Q. We can represent this by having a function AND such that AND(P,Q) denotes the conjunction of P and Q. To use McCarthy's term, AND(P,Q) is an abstract syntax for representing the conjunction of P and Q.

We will represent object language variables and constants by metalanguage constants; we will use metalanguage functions in an abstract syntax to represent object language predicates, functions, and sentence operators. For example, we will represent the object language formula $\text{KNOW}(\text{JOHN}, \exists x(P(x)))$ by the metalanguage term $\text{KNOW}(\text{JOHN}, \text{EXIST}(X, P(X)))$, where JOHN and X are metalanguage constants, and KNOW, EXIST, and P are metalanguage functions.

Since $\text{KNOW}(\text{JOHN}, \text{EXIST}(X, P(X)))$ is a term, if we want to say that the object language formula it denotes is true, we have to do so explicitly by means of a metalanguage predicate TRUE:

$\text{TRUE}(\text{KNOW}(\text{JOHN}, \text{EXIST}(X, P(X))))$.

In the possible-world analysis of statements about knowledge, however, an object language formula is not absolutely true, but only relative to a possible world. Hence, TRUE expresses not absolute truth, but truth in the actual world, which we will denote by W_0 . Thus, our first axiom

is

$$L1. \quad \forall p_1 (\text{TRUE}(p_1) \equiv T(W_0, p_1)).$$

where $T(W, P)$ means that formula P is true in world W. To simplify the axioms, we will let the metalanguage be a many-sorted logic, with different sorts assigned to different sets of variables. For instance, the variables w_1, w_2, \dots will range over possible worlds; x_1, x_2, \dots

will range over individuals in the domain of the object language; and a_1, a_2, \dots will range over agents. Because we are axiomatizing the object language itself, we will need several sorts for different types of object language expressions. The variables p_1, p_2, \dots will range over object language formulas, and t_1, t_2, \dots will range over object language terms.

The recursive definition of T for the propositional part of the object language is as follows:

$$L2. \forall w, p_1, p_2 (T(w, AND(p_1, p_2)) \equiv (T(w, p_1) \wedge T(w, p_2)))$$

$$L3. \forall w, p_1, p_2 (T(w, OR(p_1, p_2)) \equiv (T(w, p_1) \vee T(w, p_2)))$$

$$L4. \forall w, p_1, p_2 (T(w, IMP(p_1, p_2)) \equiv (T(w, p_1) \supset T(w, p_2)))$$

$$L5. \forall w, p_1, p_2 (T(w, IFF(p_1, p_2)) \equiv (T(w, p_1) \equiv T(w, p_2)))$$

$$L6. \forall w, p_1 (T(w, NOT(p_1)) \equiv \neg T(w, p_1))$$

Axioms L1-L6 merely translate the logical connectives from the object language to the metalanguage, using an ordinary Tarskian truth definition. For instance, according to L2, $AND(P, Q)$ is true in a world if and only if P and Q are both true in the world. The other axioms state that all the truth-functional connectives are "transparent" to T in exactly the same way.

AD-A162 389

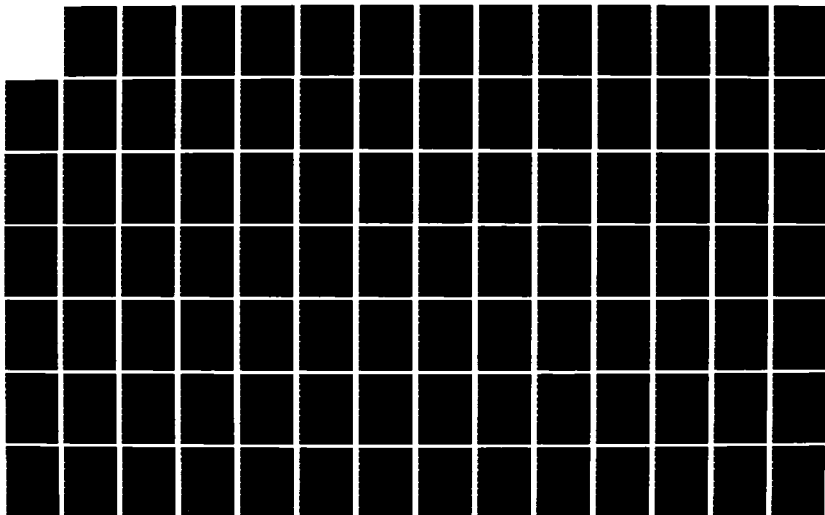
KNOWLEDGE REPRESENTATION AND NATURAL-LANGUAGE SEMANTICS
(U) SRI INTERNATIONAL MENLO PARK CA R C MOORE AUG 85
AFOSR-TR-85-1098 F49620-82-K-0031

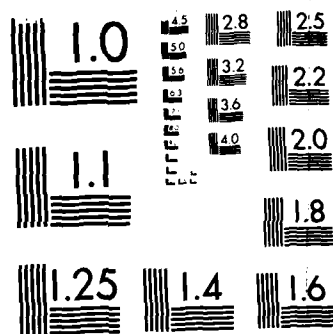
2/4

UNCLASSIFIED

F/G 5/7

NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

To represent quantified object language formulas in the metalanguage, we will introduce additional functions into the abstract syntax: EXIST and ALL. These functions will take two arguments--a term denoting an object language variable and a term denoting an object language formula. Axiomatizing the interpretation of quantified object language formulas presents some minor technical problems, however. We would like to say something like this: $\text{EXIST}(X,P)$ is true in W if and only if there is some individual such that the open formula P is true of that individual in W . We do not have any way of saying that an open formula is true of an individual in a world, however; we just have the predicate T , which simply says that a formula is true in a world. One way of solving the problem would be to introduce a new predicate, or perhaps redefine T , to express the Tarskian notion of satisfaction rather than truth. This approach is semantically clean but syntactically clumsy, so we will instead follow the advice of Scott (1970, p. 151) and define the truth of a quantified statement in terms of substituting into the body of that statement a rigid designator for the value of the quantified variable.

In order to formalize this substitutional approach to the interpretation of object language quantification, we need a rigid designator in the object language for every individual. Since our representation of the object language is in the form of an abstract syntax, we can simply stipulate that there is a function θ that maps any individual in the object language's domain of discourse into an object

language rigid designator of that individual. The definition of T for quantified statements is then given by the following axiom schemata:

$$L7. \forall w_1 (T(w_1, EXIST(X, P)) \equiv \exists x_1 (T(w_1, P[\theta(x_1)/X])))$$

$$L8. \forall w_1 (T(w_1, ALL(X, P)) \equiv \forall x_1 (T(w_1, P[\theta(x_1)/X])))$$

In these schemata, P may be any object language formula, X may be any object language variable, and the notation $P[\theta(x_1)/X]$ designates the expression that results from substituting $\theta(x_1)$ for every free occurrence of X in P.

L7 says that an existentially quantified formula is true in a world W if and only if, for some individual, the result of substituting a rigid designator of that individual for the bound variable in the body of the formula is true in W. L8 says that a universally quantified formula is true in W if and only if, for every individual, the result of substituting a rigid designator of that individual for the bound variable in the body of the formula is true in W.

Except for the knowledge operator itself, the only part of the truth definition of the object language that remains to be given is the definition of T for atomic formulas. We remarked previously that a formula of the form $P(A_1, \dots, A_n)$ is true in a world W just in case the tuple consisting of the denotations in W of the terms A_1, \dots, A_n is in

the extension in W of the relation P . To axiomatize this principle, we need two additions to the metalanguage. First, we need a function D that maps a possible world and an object language term into the denotation of that term in that world. Second, for each n -place object language predicate P , we need a corresponding $n+1$ -place metalanguage predicate (which, by convention, we will write $:P$) that takes as its arguments the possible world in which the object language formula is to be evaluated and the denotations in that world of the arguments of the object language predicate. The interpretation of an object language atomic formula is then given by the axiom schema

$$\text{L9. } \forall w_1, t_1, \dots, t_n \\ (T(w_1, P(t_1, \dots, t_n)) \equiv :P(w_1, D(w_1, t_1), \dots, D(w_1, t_n)))$$

To eliminate the function D , we need to introduce a metalanguage expression corresponding to each object language constant or function. In the general case, the new expression will be a function with an extra argument position for the possible world of evaluation. The axiom schemata for D are then

$$\text{L10. } \forall w_1, x_1 (D(w_1, Q(x_1)) = x_1)$$

$$\text{L11. } \forall w_1 (D(w_1, C) = :C(w_1))$$

$$\text{L12. } \forall w_1, t_1, \dots, t_n \\ (D(w_1, F(t_1, \dots, t_n)) = :F(w_1, D(w_1, t_1), \dots, D(w_1, t_n))),$$

where C is an object language constant and F is an object language function, and we use the ":" convention already introduced for their metalanguage counterparts.

Since $\mathcal{Q}(x_1)$ is a rigid designator of x_1 , its value is x_1 in every possible world. In the general case, an object language constant will have a corresponding metalanguage function that picks out the denotation of the constant in a particular world. Similarly, an object language function will have a corresponding metalanguage function that maps a possible world and the denotations of the arguments of the object language function into the value of the object language function applied to those arguments in that world.

It will be convenient to treat specially those object language constants and functions that are (or can be used to construct) rigid designators. We could introduce additional axioms asserting that such expressions have the same value in every possible world, but we can accomplish the same end simply by making the corresponding metalanguage expressions independent of the possible world of evaluation. So, for object language constants that are rigid designators, we will have a variant of axiom L11:

$$L11a. \quad \forall w_1 (D(w_1, C) = :C) \text{ if } C \text{ is a rigid designator.}$$

We will similarly treat rigid functions--those that always map a particular tuple of arguments into the same value in all possible worlds:

$$L12a. \forall w_1, t_1, \dots, t_n (D(w_1, F(t_1, \dots, t_n)) = F(D(w_1, t_1), \dots, D(w_1, t_n)))$$

if F is a rigid function.

Finally, we introduce a special axiom for the equality predicate of the object language, fixing its interpretation in all possible worlds to be the identity relation:

$$L13. \forall w_1, t_1, t_2 (T(w_1, EQ(t_1, t_2)) \equiv (D(w_1, t_1) = D(w_1, t_2)))$$

B. A First-Order Theory of Knowledge

The axioms given in the preceding section allow us to talk about a formula of first-order logic being true relative to a possible world rather than absolutely. This generalization would be pointless, however, if we never had occasion to mention any possible worlds other than the actual one. References to other possible worlds are introduced by our axioms for knowledge:

$$K1. \forall w_1, t_1, p_1$$

$$(T(w_1, KNOW(t_1, p_1)) \equiv \forall w_2 (K(D(w_1, t_1), w_2, w_2) \supset T(w_2, p_1)))$$

$$K2. \forall a_1, w_1 (K(a_1, w_1, w_1))$$

$$K3. \forall a_1, w_1, w_2 (K(a_1, w_1, w_2) \supset \forall w_3 (K(a_1, w_2, w_3) \supset K(a_1, w_1, w_3)))$$

K1 gives the possible-world analysis for object language formulas of the form KNOW(A,P). The interpretation is that KNOW(A,P) is true in

world W_1 just in case P is true in every world that is compatible with what the agent denoted by A in W_1 knows in W_1 . Since an object language term may denote different individuals in different possible worlds, we use $D(W_1, A)$ to identify the denotation of A in W_1 . K represents the accessibility relation associated with KNOW, so $K(D(W_1, A), W_1, W_2)$ is how we represent the fact W_2 is compatible with what the agent denoted by A in W_1 knows in W_1 .

As we pointed out before, the principle embodied in K1 is that an agent knows everything entailed by his knowledge. Since this is too strong a generalization, in a more thorough analysis we would regard the inference from the right side of K1 to the left side as being a default inference. K2 and K3 state constraints on the accessibility relation K that we use to capture other properties of knowledge. They require that, for a fixed agent A , $K(A, W_1, W_2)$ be reflexive and transitive. We have already shown this entails that anything that anyone knows must be true, and that if someone knows something he knows that he knows it. Finally, the fact that K1-K3 are asserted to hold for all possible worlds implies that everyone knows the principles they embody, and everyone knows that everyone knows, etc. In other words, these principles are common knowledge.

To illustrate how our theory operates, we will show how to derive a simple result in the logic of knowledge, that from the premises that A knows that P(B) and A knows that B = C, we can conclude that A knows that P(C). Our proofs will be in natural-deduction form. The axioms and preceding lines that justify each step will be given to the right of the step. Subordinate proofs will be indicated by indented sections, and ASS will mark the assumptions on which these subordinate proofs are based. DIS(N,M) will indicate the discharge of the assumption on line N with respect to the conclusion on line M. The general pattern of proofs in this system will be to assert the object language premises of the problem, transform them into their metalanguage equivalents, using axioms L1-L13 and K1, then derive the metalanguage version of the conclusion using first-order logic and axioms such as K2 and K3, and finally transform the conclusion back into the object language, again using L1-L13 and K1.

Given: TRUE(KNOW(A,P(B)))
TRUE(KNOW(A,EQ(B,C)))

Prove: TRUE(KNOW(A,P(C)))

- | | |
|---|-------|
| 1. TRUE(KNOW(A,P(B))) | Given |
| 2. T(W,KNOW(A,P(B))) | L1,1 |
| 3. $K(D(W,A),W,w) \supset T(w,P(B))$ | K1,2 |
| 4. $K(A(W),W,w) \supset T(w,P(B))$ | L11,3 |
| 5. TRUE(KNOW(A,EQ(B,C))) | Given |
| 6. T(W,KNOW(A,EQ(B,C))) | L1,5 |
| 7. $K(D(W,A),W,w) \supset T(w,EQ(B,C))$ | K1,6 |

8.	$K(:A(W_0), W_0, w_1) \supset T(w_1, EQ(B, C))$	L11,7
9.	$K(:A(W_0), W_0, w_1)$	ASS
10.	$T(w_1, P(B))$	4,9
11.	$:P(w_1, D(w_1, B))$	L9,10
12.	$:P(w_1, :B(w_1))$	L11,11
13.	$T(w_1, EQ(B, C))$	8,9
14.	$D(w_1, B) = D(w_1, C)$	L13,13
15.	$:B(w_1) = :C(w_1)$	L11,14
16.	$:P(w_1, :C(w_1))$	12,15
17.	$:P(w_1, D(w_1, C))$	L11,16
18.	$T(w_1, P(C))$	L9,17
19.	$K(:A(W_0), W_0, w_1) \supset T(w_1, P(C))$	DIS(9,18)
20.	$K(D(W_0, A), W_0, w_1) \supset T(w_1, P(C))$	L11,19
21.	$T(W_0, KNOW(A, P(C)))$	K1,20
22.	$TRUE(KNOW(A, P(C)))$	L1,21

A knows that $P(B)$ (Line 1), so $P(B)$ is true in every world compatible with what A knows (Line 4). Similarly, since A knows that $B = C$ (Line 5), $B = C$ is true in every world compatible with what A knows (Line 8). Let w_1 be one of these worlds (Line 9). $P(B)$ and $B = C$ must be true in w_1 (Lines 12 and 15), hence $P(C)$ must be true in w_1 (Line 16). Therefore, $P(C)$ is true in every world compatible with what A knows (Line 19), so A knows that $P(C)$ (Line 22). If $TRUE(EQ(B, C))$ had been given instead of $TRUE(KNOW(A, EQ(B, C)))$, we would have had $B = C$

true in W_0 instead of w_1 . In that case, the substitution of C for B in $P(B)$ (Line 16) would not have been valid, and we could not have concluded that A knows that $P(C)$. This proof seems long because we have made each routine step a separate line. This is worth doing once to illustrate all the formal details, but in subsequent examples we will combine some of the routine steps to shorten the derivation.

IV A POSSIBLE-WORLD ANALYSIS OF ACTION

In the preceding sections, we have presented a framework for describing what someone knows in terms of possible worlds. To characterize the relation of knowledge to action, we need a theory of action in these same terms. Fortunately, the standard way of looking at actions in AI gives us just that sort of theory. Most AI programs that reason about actions are based on a view of the world as a set of possible states of affairs, with each action determining a binary relation between states of affairs--one being the outcome of performing the action in the other. We can integrate our analysis of knowledge with this view of action by identifying the possible worlds used to describe knowledge with the possible states of affairs used to describe actions.

The identification of a possible world, as used in the analysis of knowledge, with the state of affairs at a particular time does not require any changes in the formalization already presented, but it does require a reinterpretation of what the axioms mean. If the variables w_1, w_2, \dots are reinterpreted as ranging over states of affairs, then "A knows that P" will be analyzed roughly as "P is true in every state of affairs that is compatible with what A knows in the actual state of

affairs." It might seem that taking possible worlds to be states of affairs, and therefore not extended in time, might make it difficult to talk about what someone knows regarding the past or future. That is not the case, however. Knowledge about the past and future can be handled by modal tense operators, with corresponding accessibility relations between possible states-of-affairs/worlds. We could have a tense operator FUTURE such that FUTURE(P) means that P will be true at some time to come. If we let F be an accessibility relation such that $F(W_1, W_2)$ means that the state-of-affairs/world W_2 lies in the future of the state-of-affairs/world W_1 , then we can define FUTURE(P) to be true in W_1 just in case there is some W_2 such that $F(W_1, W_2)$ holds and P is true in W_2 .

This much is standard tense logic (e.g., Rescher and Urquhart, 1971). The interesting point is that statements about someone's knowledge of the future work out correctly, even though such knowledge is analyzed in terms of alternatives to a state of affairs, rather than alternatives to a possible world containing an entire course of events. The proposition that John knows that P will be true is represented simply by KNOW(JOHN, FUTURE(P)). The analysis of this is that FUTURE(P) is true in every state of affairs that is compatible with what John knows, from which it follows that, for each state of affairs that is compatible with what John knows, P is true in some future alternative to

that state of affairs. An important point to note here is that two states of affairs can be "internally" similar (that is, they coincide in the truth-value assigned to any nonmodal statement), yet be distinct because they differ in the accessibility relations they bear to other possible states of affairs. Thus, although we treat a possible world as a state of affairs rather than a course of events, it is a state of affairs in the particular course of events defined by its relationships to other states of affairs.

For planning and reasoning about future actions, instead of a tense operator like FUTURE, which simply asserts what will be true, we need an operator that describes what would be true if a certain event occurred. Our approach will be to recast McCarthy's situation calculus (McCarthy, 1968) (McCarthy and Hayes, 1969) so that it meshes with our possible-world characterization of knowledge. The situation calculus is a first-order language in which predicates that can vary in truth-value over time are given an extra argument to indicate what situations (i.e., states of affairs) they hold in, with a function RESULT that maps an agent, an action, and a situation into the situation that results from the agent's performance of the action in the first situation. Statements about the effects of actions are then expressed by formulas like $P(\text{RESULT}(A, \text{ACT}, S))$, which means that P is true in the situation that results from A 's performing ACT in situation S .

To integrate these ideas into our logic of knowledge, we will reconstruct the situation calculus as a modal logic. In parallel to the

operator KNOW for talking about knowledge, we introduce an object language operator RES for talking about the results of events. Situations will not be referred to explicitly in the object language, but they will reappear in the possible-world semantics for RES in the metalanguage. RES will be a two-place operator whose first argument is a term denoting an event, and whose second argument is a formula. RES(E,P) will mean that it is possible for the event denoted by E to occur and that, if it did, the formula P would then be true. The possible-world semantics for RES will be specified in terms of an accessibility relation R, parallel to K, such that $R(:E, W_1, W_2)$ means that

W_2 is the situation/world that would result from the event :E happening in W_1 .

We assume that, if it is impossible for :E to happen in W_1 (i.e., if the prerequisites of :E are not satisfied), then there is no W_2 such that $R(:E, W_1, W_2)$ holds. Otherwise we assume that there is exactly one

W_2 such that $R(:E, W_1, W_2)$ holds:

$$R1. \forall w_1, w_2, w_3, e ((R(e, w_1, w_2) \wedge R(e, w_1, w_3)) \supset (w_2 = w_3))$$

(Variables e_1, e_2, \dots range over events.) Given these assumptions,

RES(E,P) will be true in a situation/world W_1 just in case there is some

W_2 that is the situation/world that results from the event described by

E happening in W_1 , and in which P is true:

$$R2. \forall w_1, t_1, p_1 (T(w_1, RES(t_1, p_1)) \equiv \exists w_2 (R(D(w_1, t_1), w_2) \wedge T(w_2, p_1)))$$

The type of event we will normally be concerned with is the performance of an action by an agent. We will let DO(A,ACT) be a description of the event consisting of the agent denoted by A performing

the action denoted by ACT. (We will assume that the set of possible agents is the same as the set of possible knowers.) We will want DO(A,ACT) to be the standard way of referring to the event of A's carrying out the action ACT, so DO will be a rigid function. Hence, DO(A,ACT) will be a rigid designator of an event if A is a rigid designator of an agent and ACT a rigid designator of an action.

Many actions can be thought of as general procedures applied to particular objects. Such a general procedure will be represented by a function that maps the objects to which the procedure is applied into the action of applying the procedure to those objects. For instance, if DIAL represents the general procedure of dialing combinations of safes, SF a safe, and COMB(SF) the combination of SF, then DIAL(COMB(SF),SF) represents the action of dialing the combination COMB(SF) on the safe SF, and DO(A,DIAL(COMB(SF),SF)) represents the event of A's dialing the combination COMB(SF) on the safe SF.

This formalism gives us the ability to describe an agent's knowledge of the effects of carrying out an action. In the object language, we can express the claim that A_1 knows that P would result from A_2 's doing ACT by saying that $\text{KNOW}(A_1, \text{RES}(\text{DO}(A_2, \text{ACT}), P))$ is true. The possible-world analysis of this statement is that, for every world compatible with what A_1 knows in the actual world, there is a world that is the result of A_2 's doing ACT and in which P is true (see Figure 5). Formally, this is expressed by

$$\forall w_1 (K(A_1, w_1, w_1) \supset \exists w_2 (R(\text{DO}(A_2, \text{ACT}), w_1, w_2) \wedge T(w_2, P))),$$

if we assume that A_1 , A_2 , and ACT are rigid designators.

In addition to simple, one-step actions, we will want to talk about complex combinations of actions. We will therefore introduce expressions into the object language for action sequences, conditionals, and iteration. If P is a formula, and ACT_1 and ACT_2 are action descriptions, then $(\text{ACT}_1 ; \text{ACT}_2)$, $\text{IF}(P, \text{ACT}_1, \text{ACT}_2)$, and $\text{WHILE}(P, \text{ACT}_1)$ will also be action descriptions. Roughly speaking, $(\text{ACT}_1 ; \text{ACT}_2)$ describes the sequence of actions consisting of ACT_1 followed by ACT_2 . $\text{IF}(P, \text{ACT}_1, \text{ACT}_2)$ describes the conditional action of doing ACT_1 if P is

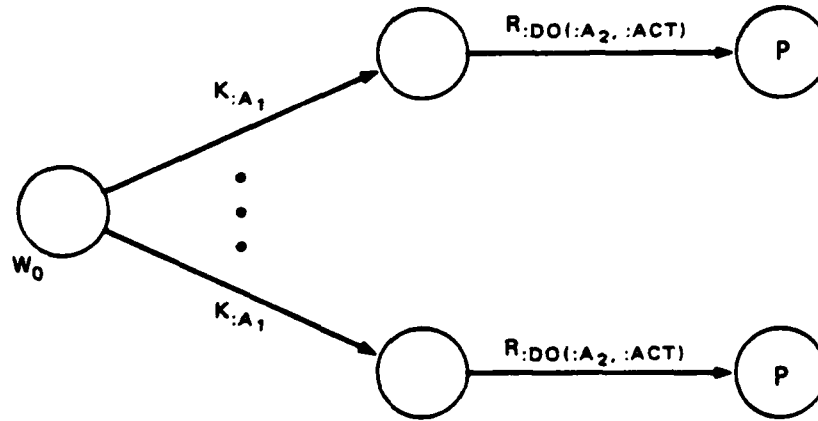


FIGURE 5 $\text{TRUE}(\text{KNOW}(A_1, \text{RES}(\text{DO}(A_2, \text{ACT}), P))) \equiv$
 $\forall w_1 (K(A_1, w_0, w_1) \supset \exists w_2 (R(\text{DO}(A_2, \text{ACT}), w_1, w_2) \wedge T(w_2, P)))$

true, otherwise doing ACT_2 . $\text{WHILE}(P, \text{ACT}_1)$ describes the iterative action of repeating ACT_1 as long as P is true.

Defining denotations for these complex action descriptions is somewhat problematical. The difficulty comes from the fact that, whenever we have an action described as a sequence of subactions, any expression used in specifying one of the subactions needs to be interpreted relative to the situation in which that subaction is carried out. For instance, if $\text{PUTON}(X, Y)$ denotes the action of putting X on Y , STACK denotes a stack of blocks, TABLE denotes a table, and TOP picks out the top block of a stack, we would want the execution of

(PUTON(TOP(STACK),TABLE); PUTON(TOP(STACK),TABLE))

to result in what were initially the top two blocks of the stack being put on the table, rather than what was initially the top block being put on the table twice. The second occurrence of TOP(STACK) should be interpreted with respect to the situation in which the first block has already been removed. The problem is that, in general, what situation exists after one step of a sequence of actions has been executed depends on who the agent is. If John picks up a certain block, he will be holding the block; if, however, Mary performs the same action, she will be holding the block. If an action description refers to "the block Mary is holding," exactly which block it is may depend on which agent is carrying out the action, but this is not specified by the action description.

One way of getting around these difficulties conceptually would be to treat actions as functions from agents to events, but notational problems would remain nevertheless. We will therefore choose a different solution: treating complex actions as "virtual individuals" (Scott, 1970), or pseudoentities. That is, complex action descriptions will not be treated as referring expressions in themselves, but only as component parts of more complex referring expressions. In particular, if ACT is a complex action description (and A denotes an agent), we will treat the event description DO(A,ACT), but not ACT itself, as having a denotation. Complex action descriptions will be permitted to occur only as part of such event descriptions, and we will define the denotations

of the event descriptions in a way that eliminates reference to complex actions. We will, however, continue to treat actions as real entities that can be quantified over, and simple action descriptions such as $DIAL(COMB(SF), SF)$ will still be considered to denote actions.

The denotations of event descriptions formed from conditional and iterative action descriptions can be defined as follows in terms of the denotations of event descriptions formed from action sequence descriptions:

$$\begin{aligned}
 R3. \forall w, t, t', t'', p \\
 & \quad \begin{matrix} 1 & 1 & 2 & 3 & 1 \\ ((T(w, p) \supset (D(w, DO(t, IF(p, t, t')))) = D(w, DO(t, t'))) \wedge \\ & \quad \begin{matrix} 1 & 1 & 1 & 1 & 1 & 2 & 3 & 1 & 1 & 2 \\ (\neg T(w, p) \supset (D(w, DO(t, IF(p, t, t'))) = D(w, DO(t, t')))) \end{matrix} \end{matrix}
 \end{aligned}$$

$$\begin{aligned}
 R4. \forall w, t, t', p \\
 & \quad \begin{matrix} 1 & 1 & 2 & 1 \\ (D(w, DO(t, WHILE(p, t))) = \\ & \quad \begin{matrix} 1 & 1 & 1 & 2 \\ D(w, DO(t, IF(p, (t; WHILE(p, t)), NIL))) \end{matrix} \end{matrix} \\
 & \quad \begin{matrix} 1 & 1 & 1 & 2 & 1 & 2 \end{matrix}
 \end{aligned}$$

R3 says that performing the conditional action $IF(P, ACT_1, ACT_2)$ results in the same event as carrying out ACT_1 in a situation where P is true or carrying out ACT_2 in a situation where P is false. R4 says that performing $WHILE(P, ACT)$ always results in the same event as $IF(P, (ACT; WHILE(P, ACT)), NIL)$, where NIL denotes the null action. In other words, doing $WHILE(P, ACT)$ is equivalent to doing ACT followed by $WHILE(P, ACT)$ if P is true, otherwise doing nothing--i.e., doing ACT as long as P remains true.

To define the denotation of events that consist of carrying out action sequences, we need some notation for talking about sequences of events. First, we will let ";" be a polymorphic operator in the object language, creating descriptions of event sequences in addition to action sequences. Speaking informally, if E_1 and E_2 are event descriptions, then $(E_1 ; E_2)$ names the event sequence consisting of E_1 followed by E_2 , just as $(ACT_1 ; ACT_2)$ names the action sequence consisting of ACT_1 followed by ACT_2 . In the metalanguage, event sequences will be indicated with angle brackets, so that $\langle :E_1 ; :E_2 \rangle$ will mean $:E_1$ followed by $:E_2$. The denotations of expressions involving action and event sequences are then defined by the following axioms:

- R5. $\forall w_1, t_1, t_2, t_3$
 $(D(w_1, DO(t_1, (t_2 ; t_3))) = D(w_1, (DO(t_1, t_2) ; DO(@D(w_1, t_3), t_3))))$
- R6. $\forall w_1, w_2, t_1, t_2$
 $(R(D(w_1, t_1), w_2, w_2) \supset (D(w_1, (t_1 ; t_2)) = \langle D(w_1, t_1), D(w_2, t_2) \rangle))$

R5 says that the event consisting of an agent A's performance of the action sequence ACT_1 followed by ACT_2 is simply the event sequence that consists of A's carrying out ACT_1 followed by his carrying out

ACT₂. Note that, in the description of the second event, the agent is picked out by the expression $\Theta(D(w_1, A))$, which guarantees that we get the same agent as in the first event, in case the original term picking out the agent changes its denotation after the first event has happened. R6 then defines the denotation of an event sequence description $(E_1 ; E_2)$ as the sequence comprising the denotation of E_1 in the original situation followed by the denotation of E_2 in the situation resulting from the occurrence of E_1 . If there is no situation that results from the occurrence of E_1 , we leave the denotation of $(E_1 ; E_2)$ undefined.

Finally, we need to define the accessibility relation R for event sequences and for events in which the null action is carried out.

$$R7. \forall w_1, w_2, e_1, e_2 \\ (R(\langle e_1, e_2 \rangle, w_1, w_2) \equiv \exists w_3 (R(e_1, w_1, w_3) \wedge R(e_2, w_3, w_2)))$$

$$R8. \forall w_1, a_1 (R(:DO(a_1, :NIL), w_1, w_1))$$

R7 says that a situation w_2 is the result of the event sequence $\langle E_1, E_2 \rangle$ occurring in w_1 if and only if there is a situation w_3 such that w_3 is the result of E_1 occurring in w_1 , and w_2 is the result of E_2 occurring

in W⁶. We will regard NIL as a rigid designator in the object language
3
for the null action, so :NIL will be its metalanguage counterpart. R8,
therefore, says that in any situation the result of doing nothing is the
same situation.

V AN INTEGRATED THEORY OF KNOWLEDGE AND ACTION

A. The Dependence of Action on Knowledge

As we pointed out in the introduction, knowledge and action interact in two principal ways: (1) knowledge is often required prior to taking action; (2) actions can change what is known. In regard to the first, we need to consider knowledge prerequisites as well as physical prerequisites for actions. Our main thesis is that the knowledge prerequisites for an action can be analyzed as a matter of knowing what action to take. Recall the example of trying to open a locked safe. Why is it that, for an agent to achieve this goal by using the plan "Dial the combination of the safe," he must know the combination? The reason is that an agent could know that dialing the combination of the safe would result in the safe's being open, but still not know what to do because he does not know what the combination of the safe is. A similar analysis applies to knowing a telephone number in order to call someone on the telephone or knowing a password in order to gain access to a computer system.

It is important to realize that even mundane actions that are not usually thought of as requiring any special knowledge are no different from the examples just cited. For instance, none of the AI problem-

solving systems that have dealt with the blocks world have tried to take into account whether the robot possesses sufficient knowledge to be able to move block A to point B. Yet, if a command were phrased as "Move my favorite block back to its original position," the system could be just as much in the dark as with "Dial the combination of the safe." If the system does not know what actions satisfy the description, it will not be able to carry out the command. The only reason that the question of knowledge seems more pertinent in the case of dialing combinations and telephone numbers is that, in the contexts in which these actions naturally arise, there is usually no presumption that the agent knows what action fits the description. An important consequence of this view is that the specification of an action will normally not need to include anything about knowledge prerequisites. These will be supplied by a general theory of using actions to achieve goals. What we will need to specify are the conditions under which an agent knows what action is referred to by an action description.

In our possible-world semantics for knowledge, the usual way of knowing what entity is referred to by a description B is by having some description C that is a rigid designator, and by knowing that $B = C$. (Note, that if B itself is a rigid designator, it can be used for C.) In particular, knowing what action is referred to by an action description means having a rigid designator for the action described. But, if this is all the knowledge that is required for carrying out the action, then a rigid designator for an action must be an executable

description of the action--in the same sense that a computer program is an executable description of a computation to an interpreter for the language in which the program is written.

Often the actions we want to talk about are mundane general procedures that we would be willing to assume everyone knows how to perform. Dialing a telephone number or the combination of a safe is a typical example. In many of these cases, if an agent knows the general procedure and what objects the procedure is to be applied to, then he knows everything that is relevant to the task. In such cases, the function that represents the general procedure will be a rigid function, so that, if the arguments of the function are rigid designators, the term consisting of the function applied to the arguments will be a rigid designator. Hence, knowing what objects the arguments denote will amount to knowing what action the term refers to. We will treat dialing the combination of a safe, or dialing a telephone number as being this type of procedure. That is, we assume that anyone who knows what combination he is to dial and what safe he is to dial it on thereby knows what action he is to perform.

There are other procedures we might also wish to assume that anyone could perform, but that cannot be represented as rigid functions. Suppose that, in the blocks world, we let $PUTON(B,C)$ denote the action of putting B on C. Even though we would not want to question anyone's ability to perform $PUTON$ in general, knowing what objects B and C are will not be sufficient to perform $PUTON(B,C)$; knowing where they are is

also necessary. We could have a special axiom stating that knowing what action PUTON(B,C) requires knowing where B and C are, but this will be superfluous if we simply assume that everyone knows the definition of PUTON in terms of more primitive actions. If we define PUTON(X,Y) as something like

```
(MOVEHAND(LOCATION(X));  
GRASP;  
MOVEHAND(LOCATION(TOP(Y))));  
UNGRASP),
```

then we can treat MOVEHAND, GRASP, and UNGRASP as rigid functions, and we can see that executing PUTON requires knowing where the two objects are because their locations are mentioned in the definition. So, although PUTON itself is not a rigid function, we can avoid having a special axiom stating what the knowledge prerequisites of PUTON are by defining PUTON as a sequence of actions represented by rigid functions.

To formalize this theory, we will introduce a new object language operator CAN. CAN(A,ACT,P) will mean that A can achieve P by performing ACT, in the sense that A knows how to achieve P by performing ACT. We will not give a possible-world semantics for CAN directly; instead we will give a definition of CAN in terms of KNOW and RES, which we can use in reasoning about CAN to transform a problem into terms of possible worlds.

In the simplest case, an agent A can achieve P by performing ACT if he knows what action ACT is, and he knows that P would be true as a

result of his performing ACT. In the object language, we can express this fact by

$$\forall a(\exists x(\text{KNOW}(a, ((x = \text{ACT}) \wedge \text{RES}(\text{DO}(a, \text{ACT}), P)))) \supset \text{CAN}(a, \text{ACT}, P)).$$

We cannot strengthen this assertion to a biconditional, however, because that would be too stringent a definition of CAN for complex actions. It would require the agent to know from the very beginning of his action exactly what he is going to do at every step. In carrying out a complex action, though, an agent may take some initial action that results in his acquiring knowledge about what to do later.

For an agent to be able to achieve a goal by performing a complex action, all that is really necessary is that he know what to do first, and that he know that he will know what to do at each subsequent step. So, for any action descriptions ACT and ACT₁, the following formula also

states a condition under which an agent can achieve P by performing ACT:

$$\forall a(\exists x(\text{KNOW}(a, ((\text{DO}(a, (x; \text{ACT}_1)) = \text{DO}(a, \text{ACT})) \wedge \text{RES}(\text{DO}(a, x), \text{CAN}(a, \text{ACT}_1, P)))) \supset \text{CAN}(a, \text{ACT}, P)).$$

This says that A can achieve P by doing ACT if there is an action X such that A knows that his execution of the sequence X followed by ACT₁ would be equivalent to his doing ACT, and that his doing X would result in his being able to achieve P by doing ACT₁.

Finally, with the following metalanguage axiom we can state that these are the only two conditions under which an agent can use a particular action to achieve a goal:

$$\begin{aligned}
 C1. \forall w, t_1, t_2, t_3, p \\
 ((t_1 = @ (D(w, t_2))) \supset \\
 (T(w, CAN(t_1, t_3, p)) \equiv \\
 (T(w, EXIST(X, KNOW(t_1, AND(EQ(X, t_2), RES(DO(t_1, t_3), p)))))) \vee \\
 \exists t_4 (T(w, EXIST(X, KNOW(t_1, AND(EQ(DO(t_4, (X; t_2)), DO(t_1, t_3)), \\
 RES(DO(t_4, X), \\
 CAN(t_1, t_3, p))))))))))
 \end{aligned}$$

Letting $t_1 = A$, $t_2 = A$, and $t_3 = ACT$, C1 says that, for any formula P,

if A is the standard identifier of the agent denoted by A, then A can

achieve P by doing ACT if and only if one of the following conditions is met: (1) A knows what action ACT is and knows that P would be true as a result of A's (i.e., his) doing ACT, or (2) there is an action

description $t_4 = ACT$ such that, for some action X, A knows that A's

doing X followed by ACT is the same event as his doing ACT and knows

that A's doing X would result his being able to achieve P by doing

ACT.

hold for all possible worlds, we are in effect assuming that it is common knowledge.

Now we show that, for any safe, if the agent A knows its combination, he can open the safe by dialing that combination; or, more precisely, for all X, if X is a safe and there is some Y, such that A knows that Y is the combination of X, then A can open X by dialing the combination of X on X:

Prove: $\text{TRUE}(\text{ALL}(X, \text{IMP}(\text{AND}(\text{SAFE}(X), \text{EXIST}(Y, \text{KNOW}(A, \text{EQ}(Y, \text{COMB}(X)))))) \text{CAN}(A, \text{DIAL}(\text{COMB}(X), X), \text{OPEN}(X))))$

1. $T(w_0, \text{AND}(\text{SAFE}(x_1), \text{EXIST}(Y, \text{KNOW}(A, \text{EQ}(Y, \text{COMB}(x_1))))))$ ASS
2. $\text{SAFE}(x_1)$ 1, L2, L9
3. $\forall w_1 (K(A(w_0), w_0, w_1) \supset (\text{C} = \text{COMB}(w_1, x_1)))$ 1, L2, L7, K1, L11, L13, L10, L12
4. $K(A(w_0), w_0, w_1)$ ASS
5. $\text{C} = \text{COMB}(w_1, x_1)$ 3, 4
6. $\text{DIAL}(\text{C}, x_1) = \text{DIAL}(\text{COMB}(w_1, x_1), x_1)$ 5
7. $T(w_1, \text{EQ}(\text{DIAL}(\text{C}, x_1), \text{DIAL}(\text{COMB}(x_1), x_1)))$ L10, L12, L12a, L13

8. $\exists w_2 (R(:DO(:A(w_0)),$ 2,D1
 $:DIAL(:COMB(w_1, x_1), x_1)),$
 $w_1, w_2) \wedge$
 $:OPEN(w_2, x_1))$
9. $T(w_1,$ L11,L10,L12a,L9,R2
 $RES(DO(\emptyset(D(w_0, A)),$
 $DIAL(COMB(\emptyset(x_1)), \emptyset(x_1))),$
 $OPEN(\emptyset(x_1)))$
10. $T(w_1,$ 7,9,L2
 $AND(EQ(\emptyset(:DIAL(:C, x_1)),$
 $DIAL(COMB(\emptyset(x_1)), \emptyset(x_1))),$
 $RES(DO(\emptyset(D(w_0, A)),$
 $DIAL(COMB(\emptyset(x_1)), \emptyset(x_1))),$
 $OPEN(\emptyset(x_1)))$
11. $K(:A(w_0), w_0, w_1) \supset$ DIS(4,10)
 $T(w_1,$
 $AND(EQ(\emptyset(:DIAL(:C, x_1)),$
 $DIAL(COMB(\emptyset(x_1)), \emptyset(x_1))),$
 $RES(DO(\emptyset(D(w_0, A)),$
 $DIAL(COMB(\emptyset(x_1)), \emptyset(x_1))),$
 $OPEN(\emptyset(x_1)))$

12. T(W₀ , 11,L11,K1

KNOW(A,
AND(EQ(@(:DIAL(:C,x₁)),
DIAL(COMB(@₁(x₁)),@₁(x₁))),
RES(DO(@₀(D(W₀,A)),
DIAL(COMB(@₁(x₁)),@₁(x₁))),
OPEN(@₁(x₁))))))

13. T(W₀ , 12,L7

EXIST(X,
KNOW(A,
AND(EQ(X,
DIAL(COMB(@₁(x₁)),
@₁(x₁))),
RES(DO(@₀(D(W₀,A)),
DIAL(COMB(@₁(x₁)),
@₁(x₁))),
OPEN(@₁(x₁))))))

14. T(W₀ , 13,C1

CAN(A,
DIAL(COMB(@₁(x₁)),@₁(x₁)),
OPEN(@₁(x₁)))

15. T(W₀ , DIS(1,14)

AND(SAFE(@₁(x₁)).
EXIST(Y,KNOW(A,EQ(Y,COMB(@₁(x₁)))))))

T(W₀ ,
CAN(A,DIAL(COMB(@₁(x₁)),@₁(x₁)),OPEN(@₁(x₁)))

16. TRUE($\text{ALL}(X,$ 15, L4, L8, L1
 $\text{IMP}(\text{AND}(\text{SAFE}(X),$
 $\text{EXIST}(Y,$
 $\text{KNOW}(A,$
 $\text{EQ}(Y, \text{COMB}(X))))))$
 $\text{CAN}(A, \text{DIAL}(\text{COMB}(X), X), \text{OPEN}(X)))$

Suppose that x_1 is a safe and there is some C that A knows to be the combination of x_1 (Lines 1-3). Suppose w_1 is a world that is compatible with what A knows in the actual world, W_0 (Line 4). Then C is the combination of x_1 in w_1 (Line 5), so dialing C on x_1 is the same action as dialing the combination of x_1 on x_1 in w_1 (Lines 6 and 7). By axiom D1, A 's dialing the combination of x_1 on x_1 in w_1 will result in x_1 's being open (Lines 8 and 9). Since w_1 was an arbitrarily chosen world compatible with what A knows in W_0 , it follows that in W_0 A knows dialing C on x_1 to be the act of dialing the combination of x_1 on x_1 and that his dialing the combination of x_1 on x_1 will result in x_1 's being open (Lines 10-12). Hence, A knows what action dialing the combination of x_1 on x_1 is, and that his dialing the combination of x_1 on x_1 will result in x_1 's being open (Line 13). Therefore A can open x_1 by dialing the combination of x_1 on x_1 , provided that x_1 is a safe and he knows the

combination of x_1 (Lines 14 and 15). Finally, since x_1 was chosen arbitrarily, we conclude that A can open any safe by dialing the combination, provided he knows the combination (Line 16).

B. The Effects of Action on Knowledge

In describing the effects of an action on what an agent knows, we will distinguish actions that give the agent new information from those that do not. Actions that provide an agent with new information will be called informative actions. An action is informative if an agent would know more about the situation resulting from his performing the action after performing it than before performing it. In the blocks world, looking inside a box could be an informative action, but moving a block would probably not, because an agent would normally know no more after moving the block than he would before moving it. In the real world there are probably no actions that are never informative, because all physical processes are subject to variation and error. Nevertheless, it seems clear that we do and should treat many actions as noninformative from the standpoint of planning.

Even if an action is not informative in the sense we have just defined, performing the action will still alter the agent's state of knowledge. If the agent is aware of his action, he will know that it has been performed. As a result, the tense and modality of many of the things he knows will change. For example, if before performing the

action he knows that P is true, then after performing the action he will know that P was true before he performed the action. Similarly, if before performing the action he knows that P would be true after performing the action, then afterwards he will know that P is true.

We can represent this very elegantly in terms of possible worlds. Suppose :A is an agent and :E₁ an event that consists in :A's performing some noninformative action. For any possible worlds W₁ and W₂ such that W₂ is the result of :E₁'s happening in W₁, the worlds that are compatible with what :A knows in W₂ are exactly the worlds that are the result of :E₁'s happening in some world that is compatible with what :A knows in W₁. In formal terms, this is

$$\forall w_1, w_2 (R(:E, w_1, w_2) \supset \forall w_3 (K(:A, w_2, w_3) \equiv \exists w_4 (K(:A, w_1, w_4) \wedge R(:E, w_4, w_3))))).$$

which tells us exactly how what :A knows after :E₁ happens is related to what :A knows before :E₁ happens.

We can try to get some insight into this analysis by studying Figure 6. Sequences of possible situations connected by events can be thought of as possible courses of events. If W₁ is an actual situation

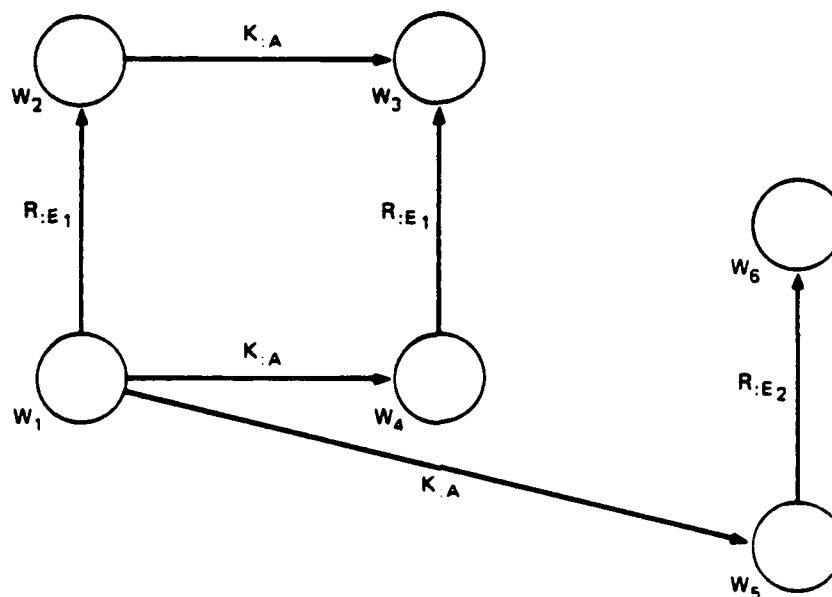


FIGURE 6 THE EFFECT OF A NONINFORMATIVE ACTION ON THE AGENT'S KNOWLEDGE

in which $:E_1$ occurs, thereby producing W_2 , then W_1 and W_2 comprise a subsequence of the actual course of events. Now we can ask what other courses of events are compatible with what $:A$ knows in W_1 and in W_2 . Suppose that W_4 and W_3 are connected by $:E_1$ in a course of events that is compatible with what $:A$ knows in W_1 . Since $:E_1$ is not informative for $:A$, the only sense in which his knowledge is increased by $:E_1$ is that he knows that $:E_1$ has occurred. Since $:E_1$ occurs at the corresponding place in the course of events that includes W_4 and W_3 ,

this course of events will still be compatible with everything :A knows in W_2 . However, the appropriate "tense shift" takes place. In W_1 , W_4 is a possible alternative present for :A, and W_3 is a possible alternative future. In W_2 , W_3 is a possible alternative present for :A, and W_4 is a possible alternative past.

Next consider a different course of events that includes W_5 and W_6 connected by a different event, :E₂. This course of events might be compatible with what :A knows in W_1 if he is not certain what he will do next. but, after :E₁ has happened and he knows that it has happened, this course of events is no longer compatible with what he knows. Thus, W_6 is not compatible with what :A knows in W_2 . We can see, then, that even actions that provide the agent with no new information from the outside world still filter out for him those courses of events in which he would have performed actions other than those he actually did.

The idea of a filter on possible courses of events also provides a good picture of informative actions. With these actions, though, the filter is even stronger, since they not only filter out courses of events that differ from the actual course of events as to what event has just occurred, but they also filter out courses of events that are

incompatible with the information furnished by the action. Suppose :E is an event that consists in :A's performing an informative action, such that the information gained by the agent is whether the formula P is true. For any possible worlds W_1 and W_2 such that W_2 is the result of :E's happening in W_1 , the worlds that are compatible with what :A knows in W_2 are exactly those worlds that are the result of :E's happening in some world that is compatible with what :A knows in W_1 , and in which P has the same truth-value as in W_2 :

$$\begin{aligned} \forall w_1, w_2 (R(:E, w_1, w_2) \supset \\ \forall w_3 (K(:A, w_2, w_3) \equiv (\exists w_4 (K(:A, w_1, w_4) \wedge R(:E, w_4, w_3) \wedge \\ (T(w_2, P) \equiv T(w_4, P))))) \end{aligned}$$

It is this final condition that distinguishes informative actions from those that are not.

Figure 7 illustrates this analysis. Suppose W_1 and W_2 are connected by :E and are part of the actual course of events. Suppose, further, that P is true in W_2 . Let W_4 and W_3 also be connected by :E, and let them be part of a course of events that is compatible with what :A knows in W_1 . If P is true in W_3 and the only thing :A learns about the world from :E (other than that it has occurred) is whether P is

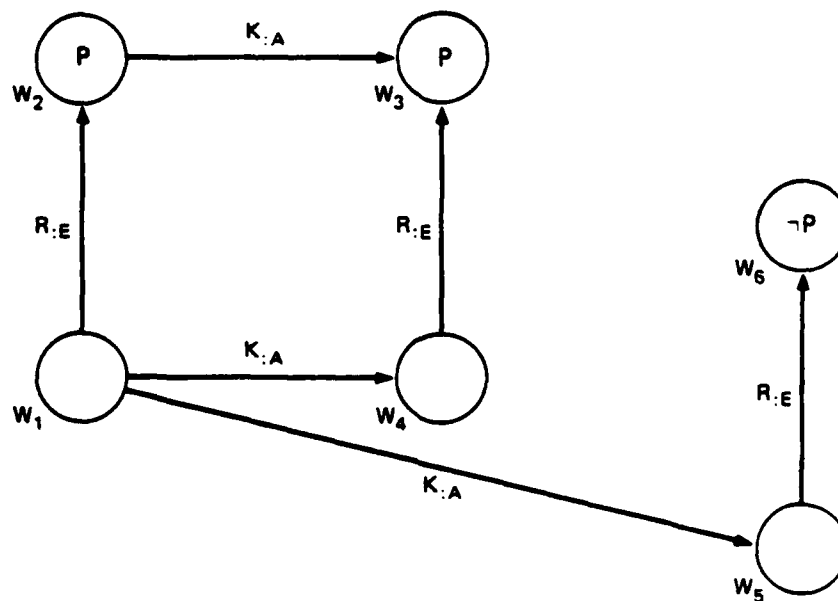


FIGURE 7 THE EFFECT OF AN INFORMATIVE ACTION ON THE AGENT'S KNOWLEDGE

true, this course of events will then still be compatible with what A knows after E has occurred. That is, W_3 will be compatible with what A knows in W_1 . Suppose, on the other hand, that W_5 and W_6 form part of a similar course of events, except that P is false in W_6 . If A does not know in W_1 whether P would be true after the occurrence of E , this course of events will also be compatible with what he knows in W_1 . After E has occurred, however, he will know that P is true; consequently, this course of events will no longer be compatible with

what he knows. That is, W will not be compatible with what :A knows in

6

W .
2

It is an advantage of this approach to describing how an action affects what an agent knows that not only do we specify what he learns from the action, but also what he does not learn. Our analysis gives us necessary, as well as sufficient, conditions for :A's knowing that P is true after event :E. In the case of an action that is not informative, we can infer that, unless :A knows before performing the action whether P would be true, he will not know afterwards either. In the case of an informative action such that what is learned is whether Q is true, he will not know whether P is true unless he does already--or knows of some dependence of P on Q.

Within the context of this possible-world analysis of the effects of action on knowledge, we can formalize the requirements for a test that we presented in Section I. Suppose that TEST is the action of testing the acidity of a particular solution with blue litmus paper, RED is a propositional constant (a predicate of zero arguments) whose truth depends on the color of the litmus paper, and ACID is a propositional constant whose truth depends on whether the solution is acidic. The relevant fact about TEST is that the paper will be red after an agent A performs the test if and only if the solution is acidic at the time the test is performed:

$(ACID \supset RES(DO(A, TEST), RED)) \wedge$
 $(\neg ACID \supset RES(DO(A, TEST), \neg RED))$

In Section I we listed three conditions that ought to be sufficient for an agent to determine, by observing the outcome of a test, whether some unobservable precondition holds; in this case, for A to determine whether ACID is true by observing whether RED is true after TEST is performed:

- (1) After A performs TEST, he knows whether RED is true.
- (2) After A performs TEST, he knows that he has just performed TEST.
- (3) A knows that RED will be true after TEST is performed just in case ACID was true before it was performed.

Conditions (1) and (2) will be satisfied if TEST is an informative action, such that the knowledge provided is whether RED is true in the resulting situation:

$T1. \forall w_1, w_2, a$
 $(R(DO(a, TEST), w_1, w_2) \supset$
 $\forall w_3 (K(a, w_1, w_3) \equiv$
 $(\exists w_4 (K(a, w_1, w_4) \wedge R(DO(a, TEST), w_4, w_3) \wedge$
 $(RED(w_2) \equiv RED(w_3))))))$

If RED and $TEST$ are the metalanguage analogues of RED and TEST, T1 says that for any possible worlds w_1 and w_2 such that w_2 is the result

of an agent's performing TEST in W_1 , the worlds that are compatible with what the agent knows in W_2 are exactly those that are the result of his performing TEST in some world that is compatible with what he knows in W_1 , and in which RED has the same truth-value as in W_2 . In other words, after performing TEST, the agent knows that he has done so and he knows whether RED is true in the resulting situation. As with our other axioms, the fact that it holds for all possible worlds makes it common knowledge.

Thus, A can use TEST to determine whether the solution is acid, provided that (1) is also satisfied. We can state this very succinctly if we make the further assumption that A knows that performing the test does not affect the acidity of the solution. Given the axiom T1 for test, it is possible to show that

$$\text{ACID} \supset \text{RES}(\text{DO}(\text{A}, \text{TEST}), \text{KNOW}(\text{A}, \text{ACID})) \text{ and } \\ \neg \text{ACID} \supset \text{RES}(\text{DO}(\text{A}, \text{TEST}), \text{KNOW}(\text{A}, \neg \text{ACID}))$$

are true, provided that

$$\text{KNOW}(\text{A}, (\text{ACID} \supset \text{RES}(\text{DO}(\text{A}, \text{TEST}), (\text{ACID} \wedge \text{RED})))) \text{ and } \\ \text{KNOW}(\text{A}, (\neg \text{ACID} \supset \text{RES}(\text{DO}(\text{A}, \text{TEST}), (\neg \text{ACID} \wedge \neg \text{RED}))))$$

are both true and A is a rigid designator. We will carry out the proof in one direction, showing that, if the solution is acidic, after the test has been conducted the agent will know that it is acidic.

7.	$\forall w_2 (K(A, w_1, w_2) \equiv$ $(\exists w_3 (K(A, w_0, w_3) \wedge$ $R(DO(A, TEST), w_3, w_2)) \wedge$ $(RED(w_1) \equiv RED(w_2))))$	5, T1
8.	$K(A, w_1, w_2)$	ASS
9.	$K(A, w_0, w_3)$	7, 8
10.	$R(DO(A, TEST), w_3, w_2)$	7, 8
11.	$RED(w_1) \equiv RED(w_2)$	7, 8
12.	$RED(w_2)$	6, 11
13.	$\neg ACID(w_3) \supset$ $\exists w_4 (R(DO(A, TEST), w_3, w_4) \wedge$ $\neg ACID(w_4) \wedge \neg RED(w_4))$	2, 9
14.	$\neg ACID(w_3)$	ASS
15.	$R(DO(A, TEST), w_3, w_4)$	13, 14
16.	$\neg RED(w_4)$	13, 14
17.	$w_2 = w_4$	15, R1
18.	$\neg RED(w_2)$	16, 17
19.	FALSE	12, 18
20.	$ACID(w_3)$	DIS(14, 19)

21.	$\text{ACID}(w_3) \supset$ $\exists w_4 (R(\text{DO}(A, \text{TEST}), w_3, w_4) \wedge$ $\text{ACID}(w_4) \wedge \text{RED}(w_4))$	1,9
22.	$R(\text{DO}(A, \text{TEST}), w_3, w_4)$	20,21
23.	$\text{ACID}(w_4)$	20,21
24.	$w_2 = w_4$	15,22
25.	$\text{ACID}(w_2)$	23,24
26.	$K(A, w_1, w_2) \supset \text{ACID}(w_2)$	DIS(8,25)
27.	$R(\text{DO}(A, \text{TEST}), w_0, w_1) \wedge$ $\forall w_2 (K(A, w_1, w_2) \supset \text{ACID}(w_2))$	5,26
28.	$\text{TRUE}(\text{RES}(\text{DO}(A, \text{TEST}), \text{KNOW}(A, \text{ACID})))$	27,L9,L11a,L12, K2,R2,L1

The possible-world structure for this proof is depicted in Figure 8. Lines 1 and 2 translate the premises into the metalanguage. Since A knows that, if the solution is acidic, performing the test will result in the litmus paper's being red, it must be true in the actual world (w_0) that, if the solution is acidic, performing the test will result in the litmus paper's being red (Line 3). Suppose that, in fact, the solution is acidic (Line 4). Then, if w_1 is the result of performing the test in w_0 (Line 5), the paper will be red in w_1 (Line 6).

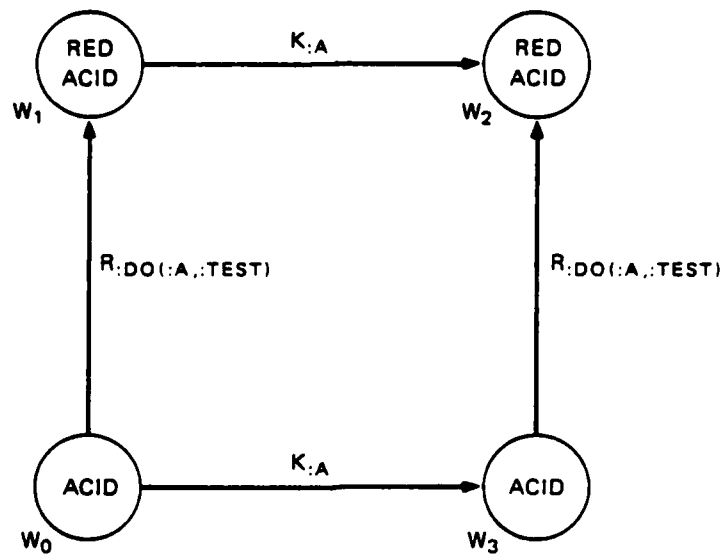


FIGURE 8 THE EFFECT OF A TEST ON THE AGENT'S KNOWLEDGE

Furthermore, the worlds that are compatible with what A knows in w_1 are those that are the result of his performing the test in some world that is compatible with what he knows in w_1 , and in which the paper is red if and only if it is red in w_1 (Line 7). Suppose that w_2 is a world that is compatible with what A knows in w_1 (Line 8). Then there is a w_3 that is compatible with what A knows in w_0 (Line 9), such that w_2 is the result of A's performing the test in w_3 (Line 10). The paper is red in w_2 , if and only if it is red in w_1 (Line 11); therefore, it is red in w_2 .

(Line 12). Since A knows how the test works, if the solution were not acidic in W_3 , it would not be acidic, and the paper would not be red, in w_2 (Line 13).

Now, suppose the solution were not acid in W_3 (Line 14). If W_4 is the result of A's performing the test in W_3 (Line 15), the paper would not be red in W_4 (Line 16). But w_2 is the result of A's performing the test in W_3 (Line 17), so the paper would not be red in w_2 (Line 18). We know this is false (Line 19), however, so the solution must be acidic in W_3 (Line 20). If the solution is acidic in W_3 , it must also be acidic in the situation resulting from A's performing the test in W_3 (Lines 21-23), but this is w_2 (Line 24). Therefore, the solution is acidic in w_2 (Line 25). Hence, in W_1 , A knows that the solution is acidic (Line 26), so in the situation resulting from A's performing the test in W_0 , he knows that the solution is acidic (Line 27). In other words (Line 28), A's performing the test would result in his knowing that the solution is acidic.

By an exactly parallel argument, we could show that, if the solution were not acidic, A could also find that out by carrying out the

test, so our analysis captures the sort of reasoning about tests that we described in Section I, based on general principles that govern the interaction of knowledge and action.

NOTES

¹ This paper presents the analysis of knowledge and action, and the representation of that analysis in first-order logic, that were developed in the author's doctoral thesis (Moore, 1980). The material in Sections III-A and III-B, however, has been substantially revised.

² Chapters 6 and 7 of (Moore, 1980) present a procedural interpretation of the axioms for knowledge and action given in this paper that seems to produce reasonably efficient behavior in an automatic deduction system.

³ "Mary's telephone number" would be an appropriate way of telling someone what John's telephone number was if he already knew Mary's telephone number, but this knowledge would consist in knowing what expression of the type "321-1234" denoted Mary's telephone number. Therefore, even in this case, using "Mary's telephone number" to identify John's telephone number would just be an indirect way of getting to the standard identifier.

⁴ This amounts to an assumption that all events are deterministic, which might seem to be an unnecessary limitation. From a pragmatic

standpoint, however, it doesn't matter whether we say that a given event is nondeterministic, or we say that it is deterministic but no one knows precisely what the outcome will be. If we treated events as being nondeterministic, we could say that an agent knows exactly what situation he is in, but, because :E is nondeterministic, he doesn't know what situation would result if :E occurs. It would be completely equivalent, however, to say that :E is deterministic, and that the agent does not know exactly what situation he is in because he doesn't know what the result of :E would be in that situation.

5

It would be more precise to say that $DO(A, ACT)$ names a type of event rather than an individual event, since an agent can perform the same action on different occasions. We would then say that RES and R apply to event types. We will let the present usage stand, however, since we have no need to distinguish event types from individual events in this paper.

6

R7 guarantees that the sequences $\langle \langle E_1, E_2 \rangle, E_3 \rangle$ and $\langle E_1, \langle E_2, E_3 \rangle \rangle$

always define the same accessibility relation on situations; so, just as one would expect, we can regard sequence operators as being associative. Thus, when we have a sequence of more than two events or actions, we will not feel obliged to indicate a pairwise grouping.

7

We have to add this extra condition to be able to infer that the

agent knows whether the solution is acidic, instead of merely that he knows whether it was acidic. The latter is a more general characteristic of tests, since it covers destructive as well as nondestructive tests. We have not, however, introduced any temporal operators into the object language that would allow us to make such a statement, although there would be no difficulty in stating the relevant conditions in the object language. Indeed, this is precisely what is done by axioms such as T1.

REFERENCES

- Frege, G. (1949) "On Sense and Nominatum," in Readings in Philosophical Analysis, H. Feigl and W. Sellars, eds., pp. 85-102 (Appleton-Century-Crofts, Inc., New York, New York).
- Hintikka, J. (1962) Knowledge and Belief (Cornell University Press, Ithica, New York).
- Hintikka, J. (1971) "Semantics for Propositional Attitudes," in Reference and Modality, L. Linsky, ed., pp. 145-167 (Oxford University Press, London, England).
- Hughes, G. E. and M. J. Cresswell (1968) An Introduction to Modal Logic (Methuen and Company, Ltd., London, England).
- Konolige, K. (1984) "Belief and Incompleteness," to appear in Formal Theories of the Commonsense World, J. R. Hobbs and R. C. Moore, eds., (Ablex Publishing Corp., Norwood, New Jersey).
- Kripke, S. A. (1963) "Semantical Analysis of Modal Logic," Zeitschrift fuer Mathematische Logik und Grundlagen der Mathematik, Vol. 9, pp. 67-96.
- Kripke, S. A. (1971) "Semantical Considerations on Modal Logic," in Reference and Modality, L. Linsky, ed., pp. 63-72 (Oxford University Press, London, England).
- Kripke, S. A. (1972) "Naming and Necessity," in Semantics of Natural Language, D. Davidson and G. Harmon, eds., pp. 253-355 (D. Reidel Publishing Company, Dordrecht, Holland).
- McCarthy, J. (1962) "Towards a Mathematical Science of Computation," in Information Processing, Proceedings of IFIP Congress 62, C. Popplewell, ed., pp. 21-28 (North-Holland Publishing Company, Amsterdam, Holland).

- McCarthy, J. (1968) "Programs with Common Sense," in Semantic Information Processing, M. Minsky, ed., pp. 403-418 (The MIT Press, Cambridge, Massachusetts).
- McCarthy, J. and P. J. Hayes (1969) "Some Philosophical Problems from the Standpoint of Artificial Intelligence," in Machine Intelligence 4, B. Meltzer and D. Michie, eds., pp. 463-502 (Edinburgh University Press, Edinburgh, Scotland).
- Moore, R. C. (1980) "Reasoning About Knowledge and Action," Artificial Intelligence Center Technical Note 191, SRI International, Menlo Park, California (October 1980).
- Quine, W. V. O. (1971) "Quantifiers and Propositional Attitudes," in Reference and Modality, L. Linsky, ed., pp. 101-111 (Oxford University Press, London, England).
- Reiter, R. (1980) "A Logic for Default Reasoning," Artificial Intelligence, Vol. 13, Nos. 1-2, pp. 81-113 (April 1980).
- Rescher, N. and A. Urquhart (1971) Temporal Logic (Springer-Verlag, Vienna, Austria, 1971).
- Russell, B. (1949) "On Denoting," in Readings in Philosophical Analysis, H. Feigl and W. Sellars, eds., pp. 103-115 (Appleton-Century-Crofts, Inc., New York, New York).
- Scott, D. (1970) "Advice on Modal Logic," in Philosophical Problems in Logic: Some Recent Developments, K. Lambert, ed. (D. Reidel Publishing Company, Dordrecht, Holland).

Appendix C

THE REPRESENTATION OF ADVERBS, ADJECTIVES AND EVENTS IN LOGICAL FORM

SRI International



THE REPRESENTATION OF ADVERBS, ADJECTIVES AND EVENTS IN LOGICAL FORM

Technical Note 344

December 1984

By: William Croft
Artificial Intelligence Center
Computer Science and Technology Division

SRI Projects 1894 and 4488

This research was supported in part by the Air Force Office of Scientific Research under Contract No. F49620-82-K-0031 and in part by the Defense Advanced Research Projects Agency under Contract No. N000039-80-C-0575 with the Naval Electronics System Command. The views and conclusion expressed in his document are those of the author and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the Air Force Office of Scientific Research, the Defense Advanced Research Projects Agency, the Naval Electronics System Command, or the U.S. Government.

ABSTRACT

The representation of adjectives and their adverbial counterparts in logical form raises a number of issues in the relation of (morpho)syntax to semantics, as well as more specific problems of lexical and grammatical analysis. This paper addresses those issues which have bearing on the relation of properties to events. It is argued that attributes and context play only an indirect role in the relation between properties and events. The body of the paper addresses the criteria for relating surface forms to logical form representations, and offers an unified analysis of adjectives and their adverbial counterparts in logical form while maintaining a clear distinction between operators and predicates; this requires the postulation of a factive sentential operator and the relaxation of the one-to-one syntax-semantics correspondence hypothesis. Criteria for determining the number of arguments for a predicate are established and are used for the analyses of phenomena such as passive-sensitivity, lexical derivational patterns, and gradability.

1 Introduction

The lexical classes "adjective" and "adverb" are distinguished in the surface structure of many natural languages, including English and the major European languages. While a fair amount of attention has been paid to the syntax and semantics of adjectives, only relatively recently have the syntax and semantics of adverbs entered the limelight. The analyses proposed for the representation of adverbs and adjectives in logical form have been quite different—partly because of the dissimilar history of such analyses in the field, but largely because they have tended to be syntax-driven; distinctions in the syntax of adjectives and adverbs have been reflected in distinctions in the logical forms proposed for them. Thus, adjectives have traditionally been analyzed as one-place predicates (or perhaps, for adjectives that take complements, as two-place predicates), since they can be predicated of noun phrases in predicative adjective constructions, and noun phrases yield arguments. Adverbs, on the other hand, have been analyzed as predicate operators, since they modify verbs or verb phrases, which are traditionally analyzed as predicates. In addition, all sentential adverbs have been analyzed as propositional operators because of a syntactic distinction between sentential and verbal adverbs.

In the past ten years or so, however, the semantics of natural language expressions, as developed by both linguists and philosophers, has freed itself more and more from a simple one-to-one correspondence with the surface syntax of English. Indeed, the easing of that constraint has enabled us to

explain some anomalous syntactic behavior. This paper will address recent semantic research in the area of adjectives and adverbs, with emphasis on its relation to the nature of events, and will argue for a more unified analysis than has previously been provided. In particular, we will argue that (1) the traditional analysis of a property as being a two-place attribute relation between an object and a value (e.g., *Color(Ford, red)*), is incorrect; (2) the proper semantic distinction is to be drawn between certain sentential adverbs, which are operators, and all remaining adverbs and adjectives, which are predicates of various types; (3) all the adverbs that have both sentential and verbal readings that are not clearly due to a lexical-semantic ambiguity can be unified in logical form as predicates; (4) the delinking of semantics from syntax extends, in the case of one subclass of adverbs, to morphology as well, and (5) many of the adjective/adverb pairs actually consist of an adjective derived from the adverb.

One of the more controversial issues in the representation of adverbs and actions will be assumed here: the need for an event variable. First proposed by Davidson [1967], this idea has been slowly but steadily growing in popularity, particularly in philosophy and artificial intelligence research. While this paper does not directly address the question of the validity of this analysis, its widespread usefulness and the unified analysis of adjectives and adverbs provided here should be taken as evidence for the analysis of events as individuals. In particular, the existence of two-place predicate adverbs, with one argument being the agent or subject of the sentence and the other the action itself, causes difficult problems for the most plausible alternative analysis of such adverbs, namely, as predicate operators.

2 Preliminaries

2.1 Lexical Semantics of Adverbs and Adjectives

Before analyzing the logical form of adjectives and adverbs, henceforth referred to as AA's, I shall list the major lexical semantic classes of adverbs that are relevant to this study, and the names for these classes that have been used in the literature. Besides serving to delimit the range of our study, this classification will provide a basis for the semantic issues to be discussed subsequently. This is not intended to be an exhaustive list of the lexical semantic classes that fall under the logical forms to be presented here; it is, however, a superset of the lexical classes of AA's whose semantic behavior has been discussed in the literature. Terms used by other authors are shown

in parentheses.

1. Operators

- (a) Modal ([Bellert 1971]; Epistemic [Ernst 1984a]): *possibly, probably, necessarily, not*, etc.
- (b) Evidential (Epistemic [Ernst 1984a]; Modal [Bellert 1971]): *evidently, obviously, allegedly, presumably*, etc.

2. Predicates

- (a) Two-place predicates [arguments for agent and event, proposition, etc.]
 - i. Behavior (Agent-Oriented [Ernst 1984a]; $P_{subject}$ [Jackendoff 1972]): *rudely, nicely, politely*, etc.
 - ii. Ability (Agent-Oriented [Ernst 1984a]; $P_{subject}$ [Jackendoff 1972]): *cleverly, foolishly, stupidly*, etc.
 - iii. Intentional (Volitional [Ernst 1984a]; Passive-sensitive [McConnell-Ginet 1981]): *intentionally, willingly, reluctantly*, etc.
 - iv. Evaluative (also [Ernst 1984a], [Bellert 1977]; $P_{speaker}$ [Jackendoff 1972]): *fortunately, surprisingly, luckily, oddly*, etc.
 - v. Derived two-place Measure terms [see Section 3.4]
- (b) One-place predicates
 - i. Emotional State (Mental State [Ernst 1984a]): *bitterly, angrily, gloomily, furiously*, etc.
 - ii. Measure
 - A. Normal: *successful(ly), beautiful(ly), good/well, tall, thin, short, slow, quick*, etc.
 - B. Facility: *easy, tough, simple, difficult*, etc.
 - iii. Qualitative: *red, black, dark, square*, etc.

[Note: Measure terms and other gradable AA's also have arguments for the reference set, as well as perhaps for the quantity or degree.]

There are a number of phenomena, labeled "adverbial" in the literature, that will not be discussed here. Of these, the most important are words, phrases, and clauses that refer to the time or location of an event. While these are clearly sentential adverbs in their behavior, current proposed extensions or modifications of first-order logic have specific ways of accounting

for time and location of events which are independent of the logical issues to be examined in this paper. The other major class of "adverbs" that will not be addressed comprises such verbal arguments as Instrument, Source and Goal, which have been called adverbs in the linguistic literature presumably because, unlike subject, object, and indirect object, they are syntactically optional, but which are clearly arguments of the appropriate verbal predicates.

There is a third class of adverbs that will also be disregarded in this paper: those that are derived from nouns and mean (to use the classic dictionary definition) "in some manner of, related, or pertaining to X", such as *electrically* in *electrically charged* or *electrically activated*. These are instances of the same kind of context-specific meaning relation as complex nominals, i.e., such constructions as *circuit board*, *syntax class*, etc. It has been demonstrated [Levi 1978] that adjectival forms derived from nouns that mean "of, related, or pertaining to X" behave syntactically and semantically like complex nominal constructions, and just happen to be syntactically adjectivalized because they are functioning "like" adjectives. Likewise, the denominal adverbs such as *morphologically* and *electrically*—like other adverbs with adjectival counterparts—take the adverbial morphology because they are functioning as modifiers of verbs or adjectives, a strictly syntactic fact.

2.2 The Status of Attributes

There is a long-standing philosophical tradition stretching back to at least Aristotle that treats properties (color, shape, size, etc.—the basic, "core" adjective concepts) as values of an attribute of the object rather than as directly predicated of objects themselves. Thus, *The box is red* would be analyzed as something like **Color(Box, Red)**—or, more abstractly, **Attribute(Box, Color, Red)** rather than simply **Red(Box)**. This analysis of properties and attributes has also been used extensively by those artificial intelligence traditions that employ "semantic nets" and "frames" [Woods 1975:50]. While this analysis is rather inelegant, it does appear to account for two constraints on adjective behavior. Adjectives (and adverbs as well [Bresnan 1982:164-65]) are usually considered to be recursive in the syntax; an arbitrarily great number of them can appear as modifiers of a single noun. There are two constraints on their (co)occurrence: they must be values of an attribute that the object denoted by the head noun possesses (e.g., **a red electron* is unacceptable), and no more than one can occur modifying

the same attribute (e.g., **a purple magenta book*, meaning a book that is both purple and magenta, rather than one whose color is a cross between purple and magenta, is unacceptable). The value-as-argument analysis of properties allows one to capture these constraints quite easily, while the value-as-predicate analysis does not seem to do so at all.

There is, however, an interpretation of attributes and values that allows us to maintain a logical form that does not explicitly represent the attribute, retain the value-as-predicate analysis, and nevertheless be able to account for the aforementioned constraints. Various English constructions support analysis of an attribute's values as belonging to a lower-level type, while the attribute itself is a higher-level type subsuming the attribute's values. Consider the following sentences:

- (1) The book is red.
- (2) Fido is a pug.
- (3) Red is a color.
- (4) The pug is a dog.
- (5) My jacket is the same color as your book; it's maroon.
- (6) That is the same dog as mine: it's a pug.

The adjective and attribute-name uses in the odd-numbered examples above are parallel to the even-numbered noun uses just below them. Examples 1-4 all use the "be of predication", which takes an individual (1-2) or a lower-level type (3-4) as the subject and an expression representing a type or a kind higher than that of the subject as the predicate (supported by the copula). Thus, in 1 *red* functions as a type, while in 3 *color* functions as a type higher than *red*.¹ The examples in 5-6 all use the "be of identity", asserting the equivalence of a type lower than *color* or *dog*, since it is obviously not being asserted that the two individuals themselves are identical. In 5, the lower level type is the value, *maroon*, which is exactly parallel to the lower-level type *pug* in 6.

If we adopt the analysis implied in the examples, i.e. that attributes constitute a higher-order type, then the two constraints discussed earlier emerge automatically from the standard behavior of type hierarchies. An individual cannot be a member of two disjoint sister sets at the same time; thus **a purple magenta book* is parallel to **a dog that is a cat*. Likewise,

¹Predicate adjectives are also subject to a syntactic constraint against taking articles and plurals, thus resembling mass terms instead of count terms like *pug* or *dog*; a better example than 4 would be *Water is a liquid*.

an individual can be a member only of supersets of the basic set, so **a red electron* is parallel to **a dog that is a crime*.

Another aspect of attributes that suggests they should be left out of the logical form of AA's is their predictability. Unlike such phenomena as reference sets for measure terms, which have been shown to vary unpredictably and require an additional argument position in the predicate type (see footnote 14), the attribute is predictable from the value provided. The only exceptions to this rule are such value terms as *green*, which are ambiguous across attribute values—in this example, color vs. ripeness vs. emotional state vs. experience. In these cases, the ambiguity is always finite and lexically fixed, and so is of a completely different order of complexity from the reference set example.

Everything that has been said above concerning adjectives can also be stated *mutatis mutandis* with regard to verbal adverbs. These adverbs are analyzed as modifying an event variable, which can be thought of as a variable that describes an event or, more precisely, a process. Here again, [verbal] adverbs can be applied indefinitely to verbs, subject to the two constraints given above, and the attributes involved (result, direction, speed, etc. of the process) are actually higher-level types.

There is, however, one feature of the adjective-noun relation that is a priori unpredictable and requires context or world knowledge to disambiguate; this feature resembles that of complex nominal expressions such as *book department* or *glare screen*, in which the exact relation between the head and the modifier is left unspecified until the context can make it more precise [Downing 1979]. If one compares the phrases *a red apple* and *a mushy apple*, it is immediately evident what attribute is assumed in each case, i.e., color and texture, respectively—but the first attribute pertains to the surface of the apple, while the second pertains to its interior. In both cases, general world knowledge about the structure of apples and about which attributes of which parts of apples are most relevant to people determines that we are not dealing with a red-fleshed apple or one whose skin resembles foam rubber; on the contrary, this knowledge is both object- and context-specific. This leads to ambiguities that are potentially indefinitely large, just as with noun modifiers. Consider the following example (used by John McCarthy in a seminar at Stanford to make a similar point): *red* in *red pencil* could refer to the color of the pencil's surface, or to the color of the mark left after the pencil has been used to write or draw, or (in theory) to any other part or aspect of the pencil or its function which the speaker finds salient enough to describe. The chief difference between adjectival modifiers and

noun modifiers is that, in most cases, the part or aspect of the object that is appropriately described by the adjective is almost always determined by general knowledge about the object itself, the specific situational context contributing relatively little; on the other hand, the precise relation between the noun modifier and its head is established at least as much by the specific context of use as by our general knowledge. This aspect of adjectival behavior must be treated the same way as the corresponding behavior of noun modifiers. Thus, technically, any predication of a property should be of the form $\text{Adj}(\mathbf{F}(\mathbf{x}))$, in which \mathbf{F} is a context-determined function from the entity \mathbf{x} to the part or aspect of the entity that Adj is really a property of, just as a complex nominal form $[\mathbf{x} \mathbf{y}]_N$ is really $\mathbf{R}(\mathbf{x}, \mathbf{y})$, in which \mathbf{R} is a context-determined relation that is the exact relation between the two entities. This added notational necessity is acknowledged here, but will be disregarded in the rest of this paper.²

3 Logical Types for Adverbs and Adjectives

3.1 Modal Adverbs: The Thomason and Stalnaker Tests

As stated above, the principal line to be drawn between classes of AA's at the level of logical form is between operators and predicates. The classic examples of operator adverbs are those that correspond to the modal operators: *possibly*, *necessarily*, and the sentence negator *not*. In addition, it is incontrovertible that the evidential adverbs such as *probably* and *evidently* are also sentence operators. The evidential adverbs all reflect different degrees of knowing something, in particular degrees of uncertainty of knowing something; therefore, under the possible worlds interpretation of knowledge

²It seems that the irregular semantic behavior of nouns and adjectives is associated with some characteristic of nouns themselves. All of those cases described in the literature in which compositional and referential semantics must take world knowledge and/or the specific context prominently into account have to do with nouns. In addition to the irregular compositionality in the syntax of adjective-noun and noun-noun constructions mentioned in the text, there is an irregular compositionality in the morphology associated with denominal derivations that is not found with deverbal or deadjectival derivations. Thus, for example, denominal verbs are highly irregular in their semantics; what Clark and Clark [1979] show for zero derivation is also true for nonzero derivation—compare *colonize*, *alphabetize*, *atomize*, or the innovation *productize*). The same is true of denominal agentive nouns: compare *scientist*, *machinist*, *violinist*, *communist*. Finally, as Geoffrey Nunberg has amply demonstrated [Nunberg 1979], simple nominal reference per se is also highly sensitive to world knowledge and context of situation.

and belief [Hintikka 1971], they are parallel to the modal operators.

Thomason and Stalnaker [1973] propose four criteria for deciding whether an adverb is sentential or not. Although they consider each test to be a sufficient condition in itself, a detailed study of individual adverbs indicated that, in most cases, all four conditions applied if any one did. More important to the current line of research is the fact that three of the four criteria test specifically for behavior that characterizes modal operators, at least in the possible worlds interpretation of modality. The first criterion is whether or not the adverb induces referential opacity in the entire sentence. While referential opacity is not unique to modal contexts and the like, it is characteristic of all of them. The same is true for scope ambiguity, the property used in the second criterion. Scope ambiguity is a feature of quantifiers as well as modal operators; however, in the possible worlds interpretation of modality, the basic modal operators behave like quantifiers over possible worlds. The third semantic criterion is whether or not the adverb is semantically appropriate in the context *It is Adv true that S*. In the sense that operators apply propositions to possible worlds and truth is defined as the applicability of a proposition in a world (i.e., truth is relativized to "truth in a world"), this criterion also is a criterion for operator status.³ The remaining criterion, namely, that an adverb is sentential if it outscopes an adverb already proved to be a sentential modifier, is syntactic in nature and appears to be inessential, since, in all of the cases considered, the other criteria sufficed.⁴

³This test is closely related to a syntactic property of sentential adverbs, namely, that they can be paraphrased with their adjectival counterparts in the construction *It is Adj that S*. This fact places the adjective *likely* in the Evidential class—which its lexical semantics would certainly indicate—although, apparently for phonological reasons, it has no adverbial counterpart.

⁴There are some uses of Modal and Evidential AA's as adjectives modifying single nouns or fragments of noun phrases: *the alleged killer of the child*, *a possible solution*, etc. The meaning of these phrases can be paraphrased as *the person who is allegedly the killer of the child* and *a thing that is possibly a solution*; in logical form, this would simply be represented as an operator having scope over the relevant conjunction of predicates (represented here in a restricted quantification notation): [**the x: Alleged(Kill(x, child))**] and [**an x: Possible(Solution(x))**]. A similar analysis would be required for another subclass of adverbs: *hopefully*, *ideally*, and *desirably*, first noted by Ernst [1984a:71-73]; they would have to be modal operators over the entire sentence. Finally, adjectives like *fake*, *toy*, and *imitation* seem to require analysis as true predicate operators, since they alter the meaning of the predicate rather than the possible-worlds (i.e. epistemological/mental) status of the proposition. However, all the proposed operators—both sentential and predicate—have in common the fact that the truth of

3.2 S/V Adverbs and the FACT Operator

The discussion concerning Thomason and Stalnaker's criteria and its reference to the nature of operators (or rather, the shared properties of concepts that are represented as operators in logical form) highlights the problem of adverbs which appear to be ambiguous between sentential and verbal readings (S/V adverbs). If we adopt the interpretation of events as an independent argument of a predicate, as advocated by Davidson [1967], Moore [1981] and others, and argued for extensively by McConnell-Ginet [1981], then verbal adverbs will be predicates on that event variable. However, if there are adverbs that have both a verbal and a sentential reading (the latter proved by means of Thomason and Stalnaker's criteria), then we appear to be faced with one of two unpleasant alternatives: either to say that there are two otherwise synonymous terms, one an operator and the other a predicate, or that the single term is of one type, thereby forcing all verbal adverbs to be operators. Fortunately, there is a solution to this problem that reveals a "hidden" operator whose existence is supported by independent linguistic evidence.

Let us consider the example of Behavior adverbs such as *rudely* and *politely* and Ability adverbs such as *cleverly* and *stupidly*, etc. in which the distinction between the sentential and verbal readings is clearest. The following pairs of sentences, otherwise identical except for the position of the adverb, mean distinct things:

- (7) Maggie spoke rudely to the Queen.
- (8) Rudely, Maggie spoke to the Queen.
- (9) Jerry opened the window cleverly.
- (10) Cleverly, Jerry opened the window.

In the first sentence of each pair, the action was performed in a manner that is described by the adverb: it was perhaps Maggie's tone of voice or her use of brusque language that made the event rude, while it was presumably Jerry's technique in opening the window that was clever. This is clearly a verbal reading, with the predicate modifying the event variable. In the second sentence, it is the performance of the act itself (as opposed to its nonperformance) that is described by the adverb: Maggie was rude to speak to the Queen, while Jerry was clever to open the window at that time.

$P(x)$ does not immediately follow from $OP[P(x)]$ or $[OP(P)](x)$. While this is a general property of operators, it is not, as we shall see, a necessary one.

The readings in 8 and 10, generally called "sentential" readings due to their syntactic behavior, pose difficulties in analysis because they do not seem to fulfill Thomason and Stalnaker's semantic criteria for sentential adverbs. The sentential readings do not induce opacity in the sentence, the first criterion; in fact, unlike most other sentential adverbs, they are factive. When Thomason and Stalnaker's second criterion is applied, as in 11 and 12 below, one finds distinct readings under an interpretation in which one person speaking to the Queen is acceptable, but everyone speaking to the Queen at once is not:

- (11) Everyone rudely spoke to the Queen.
- (12) Rudely, everyone spoke to the Queen.

However, the phenomenon in 11 and 12 has a different explanation that is independent of the verbal vs. sentential adverb distinction. In another part of their paper, Thomason and Stalnaker [1973:200] point out that sentences like 13 and 14, with the adverb *slowly*—about as impeccable a verbal adverb as one can find—also display "scope ambiguity":

- (13) Slowly, everyone left.
- (14) Everyone left slowly.

In this case, as in 11 and 12, the "adverb wide scope" reading is actually a predication of the adverb over a distinct kind of event, i.e., the event of a collective group doing X, which happens to look like an aggregate of individual doing-X events. The property denoted by the adverb applies to that collective event (the slowness of everyone viewed as a group to leave, the rudeness of everyone viewed as a group to speak to the Queen). Thus, the phenomenon in 11 and 12 do not qualify as support for Thomason and Stalnaker's criterion.

Finally, the third criterion, acceptability in the frame *It is Adj that S*, does not appear to apply; 15 and 16 are not especially good English:

- (15) *?It was rudely true that Maggie spoke to the Queen.
- (16) *?It was cleverly true that Jerry opened the window.

Thomason and Stalnaker themselves argue that locative and temporal adverbs satisfy their third criterion, adducing 17 and 18 as evidence (Thomason and Stalnaker [1973:206]), but these examples are no more convincing than 15 or 16:

- (17) *?It is true in the morning that Mary beats her dog.
(18) *?It was true in the kitchen that Henri dropped the souffle.

The sentential readings in 8 and 10 are characterized in a number of ways. First, unlike most sentential adverbs, they are factive. Second, it is just this factivity that the truth conditions for the sentential adverb reading are sensitive to: thus, it is the fact that the event in question falls under the description of "Maggie speaking to the Queen" that makes it rude. Nevertheless, the meaning of the adverb *rudely* (as well as *cleverly* and the like) is the same in both the verbal and sentential readings.

One must not confuse the meaning of utterances like 7-10 with explanations as to why the action, or its execution, is rude, clever, etc. Earlier proposals for analyzing 7 and 8 suggested that the difference between the verbal and sentential reading was that in 8 it was the fact that the action fitted the description provided by the proposition that made it rude, whereas in 7 it was some other description of the action (speaking loudly, using obscenities, etc.) that made it rude. However, what made the action of Maggie's speaking to the Queen rude in 8 may have to do with all sorts of things that may be quite remotely linked to the description. First, it may be that only part of the description is relevant to the reason for the action—e.g., the act of speaking to the Queen, not that of Maggie's speaking to the Queen. Or, conversely, it was only in the given context—not at all mentioned in the proposition under the "scope" of the adverb—that Maggie's speaking to the Queen was rude. The important point is that all sentence 8 asserts is that the fact that that event happened under those circumstances, as opposed to its not happening at all or to some other event's happening, was rude. Any inference as to the reason the fact that that event occurred was rude is not part of the semantics of 8. Likewise with 7: only some property of the event rather than its existence is asserted to be rude; the question what that property was or why it is considered to be rude is left open.

The verbal/factive-sentential ambiguity phenomenon appears to be present in all of the two-argument (actor and event) adverb lexical classes except for the Intentional class, and is usually the only reading available for the Evaluative subclass. The Emotional State adverbs such as *angrily*, whose semantics means roughly "x is such that one can infer that the agent was angry", has two distinct readings:

- (19) Sue shut the door angrily.
(20) Angrily, Sue shut the door.

The preferred reading in 19 is that the manner in which Sue shut the door implied anger on her part, while the preferred reading for 20 is that the fact that Sue shut the door (say, the door to a dorm room during a hall party), as opposed to not doing so, indicated that she was angry. Even though both readings are possible in either position, the positional preferences for English adverbs merely tend to suggest the sentential or verbal readings for those adverbs that have both.⁵

Ernst [1984a] considers the possibility that Intention adverbs also display both readings:

- (21) Sue closed the door deliberately.
- (22) Sue deliberately closed the door.

There may be a reading of 22 that means that the manner in which Sue closed the door was deliberate on her part, while in 18 Sue's intention was to close the door; in addition, the sentence indicates her successful accomplishment of the act (the more common reading). If 22 is indeed a verbal adverb, it must be a derived one because it does not display the other behavior of Intentional adverbs, such as the opacity of the VP (see below).

Finally, with Evaluative adverbs like *fortunately* or *luckily*, as in 23, it is clearly the fact of Sue's shutting the door that is fortunate or lucky, not the manner in which she did it:

- (23) Fortunately/Luckily, Sue shut the door.

However, some of the Evaluative adverbs do allow a verbal adverb reading, as noted by Ernst ([1984a:66], his examples 169 and 173):

- (24) That performance turned out pretty luckily, considering all the trouble we had beforehand.
- (25) Joan thought Fenster would be elated, but he reacted very curiously/strangely to the news.

Such examples are extremely rare, however.⁶

⁵The fact vs. manner distinction may not be present in the semantic representation of utterances with Emotional State adverbs; it may be only a part of the *reason* the agent was angry, etc., and so the arguments in the preceding paragraph apply. The lexical semantics of Emotional State adverbs appears to be vague rather than ambiguous with respect to the fact/manner distinction. See also the discussion of Emotional States AA's in Section 3.4.

⁶The factive readings of Evaluative adverbs, unlike those of other AA's, allow the para-

The solution to the dilemma of how to represent the semantic unity of the predicates that have both sentential and verbal adverb readings is to realize that there are two different things being characterized in the members of each pair.⁷ The first is an event in the world, which is represented by the event variable. The second and more abstract one is the state of affairs of that proposition's being true. This, like an event, is part of the world; but, unlike events, it is something associated with every [true] proposition. This corresponds to the paraphrase of sentence 8 as *The fact that Maggie spoke to the Queen was rude*; the fact that Maggie spoke to the Queen is as much part of the world as the event that happened to be an instance of Maggie's speaking to the Queen. Indeed, the best way to test for the the sentential adverb reading of a predicate is to see whether the paraphrase *The fact that S is Adj* makes sense. To put it in terms suggestive of situation semantics [Barwise and Perry 1983], the state of affairs is the [factual] existence of something subsumed under a complex event type, e.g., "Maggie speaking to the Queen". No part of the description of the event is dispensable for the factive reading; still, for the reasons indicated above, one cannot draw any inference outside context as to what aspect or circumstance of the described event furnishes a rationale for the event's being rude or the like.

There is further evidence that supports this hypothesis. Adverbs like *rudely* or *cleverly* in their sentential readings (and also adverbs of the Evaluative class), can be applied to any sentence, including stative sentences. In the latter, however, the second, verbal adverb reading is absent—precisely because there is no event variable present. Thus, 26 has only one reading (the sentential one) and 27 is unacceptable because the sentential reading (the only possible one) is not possible with the adverb immediately following the main verb:

(26) Rudely, Fred was late to the Presidential dinner.

(27) *Fred was late rudely to the Presidential dinner.

Another prediction that one would make from the hypothesis is that

phrase *It is Adj that S*, e.g. *It is fortunate/lucky that Sue shut the door* or **It was clever that Jerry opened the window*; the nearest acceptable paraphrase for the latter classes requires the presence of the subject of the infinitive form in a PP: *It was clever of Jerry to open the window*. The reason for this appears to be that whereas in all the other AA classes the second argument to the predicate must be a participant in the action, this semantic restriction does not apply to Evaluative AA's (see section 3.4).

⁷This analysis was proposed by Robert Moore, in the course of discussions of this paper with the author.

the adverbs that are genuine operators, namely, the Modal and Evidential adverbs would not have any sort of verbal adverb readings with the same meanings. This prediction is also correct, as Ernst [1984b] has observed: in the case of those Evidential adverbs that do appear to have verbal adverb counterparts, the latter actually have meanings that differ from the corresponding sentential readings:⁸

(28) Clearly, John is right.

(29) John spoke clearly.

Furthermore, these classes of adverbs are not the only linguistic phenomenon to exhibit this semantic ambiguity. Such factive predicates as 30, first discussed by Kiparsky and Kiparsky [1970], also have two readings corresponding to those of *rudely* and *cleverly*, which are paraphrased in 31 and 32:

(30) Mary disapproves of John's drinking.

(31) Mary disapproves of the way John drinks.

(32) Mary disapproves of the fact that John drinks.

Finally, states of affairs, as well as events, enter into causal relations, so that the situation in 33a is described by 33b; note that no event variable could be involved, since the causal clause in 33a is stative. On the other hand, 34a exhibits both the manner and fact readings:

(33a) The President's being late caused the banquet to be delayed for two hours.

(33b) The fact that the President was late caused the banquet to be delayed for two hours.

⁸The only possible exception to this rule seems to be *obviously*, which has a verbal adverb counterpart with a lexical semantics that does not appear to be distinct from the evidential form:

(a) Obviously, someone opened the door.

(b) Sandy opened the door obviously.

Sentence *b* means roughly "Sandy opened the door in a manner that made her action obvious", in the evidential sense of *obvious*. This was first pointed out by Ernst: "While a unified sense...works for *obviously*, it seems that no other Epistemic [Evidential] adverb admits of such treatment" [Ernst 1984b:87]. Unless a semantic difference between the two readings of *obviously* is found, this adverb may be a counterexample to our proposal.

- (34a) John's drinking makes Mary upset.
- (34b) The way John drinks makes Mary upset.
- (34c) The fact that John drinks makes Mary upset.

Indeed, any natural language expression (nominalizations as well as complements) that can be paraphrased with *the fact that S* without altering the truth conditions of the utterance will be subject to the same kind of analysis as the phenomena described above.

All of this evidence confirms that a general systematic phenomenon is occurring here. The fact that the sentential readings exhibit the semantic behavior tested by Thomason and Stalnaker suggests that the "fact" reading should be characterized by an operator, which we will call **FACT**, which has scope over the proposition, and which denotes a function from the latter to a state of affairs. Hence, the two readings embodied in 7 and 8 would be represented as follows (**Rude** is a two-place predicate):

- (35) $\exists e[\text{Speak}(e, \text{Maggie}, \text{Queen}) \ \& \ \text{Rude}(\text{Maggie}, e)]$
- (36) $\exists e[\text{Speak}(e, \text{Maggie}, \text{Queen}) \ \& \ \text{Rude}(\text{Maggie}, \text{FACT}(\text{Speak}(e, \text{Maggie}, \text{Queen})))]$.

3.3 Adverbs of Intention

There is one class of two-place predicate AAs, referring to mental states, that behaves distinctly from all the other AA classes, namely, the Intentional class. The adverbs of this class do not have the S/V distinction, they induce opacity, and they display "passive-sensitivity" ([McConnell-Ginet 1981:145; see below).

The distinctive behavior of the Intentional class of adverbs can be largely explained by treating them in a manner parallel to that applied to the verbs from which they are derived or to which they are related—i.e. verbs that denote intention, desire and knowledge, that have a proposition as one of the arguments of the predicate. Thus, just as with the Modal and Evidential adverbs, the S/V distinction is not relevant to the Intentional class. Like the lexically and semantically related verbs and adjectives of intention etc., the adverbs induce opacity:

- (37) George intentionally/willingly attacked Ronald Reagan.
- (38) George intended/was willing to attack Ronald Reagan.
- (39) Ronald Reagan is the President of the United States.

- (40) \nV George intentionally/willingly attacked the President of the United States.
- (41) \nV George intended/was willing to attack the President of the United States.

In a situation in which George did not know that Ronald Reagan was the President of the United States, 40/41 do not follow from 37/38 and 39.

Unlike the Behavior and Ability adverbs, the corresponding verbal or adjectival forms of Intentional adverbs are not factive:

- (42) Harvey was willing to cut the roast \nV Harvey cut the roast.
- (43) Harvey was stupid to cut the roast before cooking it \vdash Harvey cut the roast before cooking it.

The Intentional adverb forms themselves are factive (e.g., 37), indicating that (like all other adverbs, except the Modal and Evidential ones, and like most adjectives as well) two assertions are involved. Finally, like the corresponding verbal forms but unlike the Modal and Evidential adverbs, the Intentional adverbs take a second argument: the participant who intended, was willing, etc., to perform the action he has performed. Therefore, to capture all of these semantic facts, a logical form for 37 would have to be the one in 44; compare 45, which is the logical form of 38:

- (44) $\exists e[\text{Attack}(e, \text{George}, \text{RR}) \ \& \ \text{Intend}(\text{George}, \text{Attack}(e, \text{George}, \text{RR}))]$
- (45) $\text{Intend}(\text{George}, \exists e[\text{Attack}(e, \text{George}, \text{RR})])$

It is worth noting at this point that an anomaly in the interpretation of *intentionally* provides an additional piece of evidence for the existence of an event variable (as suggested by Robert Moore [personal communication]). Let us consider the following situation, taken from Searle ([1980:51]; in turn borrowed from Chisholm [1966]): John intends to kill his uncle, in order to collect early on his inheritance. He gets into his car to drive to his uncle's house, but in his haste to get there he runs over an old man—who, unbeknownst to John, is his uncle. Question: did John intentionally kill his uncle? If the standard notation without the event variable as in 46 is used, then the answer is yes, since there is no way to indicate that the killing of his uncle in the first conjunct is the same action as in the second conjunct, i.e., that John intended that very event to be the killing of his uncle. John clearly did not intend the event of the car accident to be the event of his killing his uncle—he had something completely different in

mind—and so the traditional representation makes an erroneous prediction. However, the representation that includes the event variable in 47 does make the correct prediction, because the identity of the event variable in the second conjunct with the one in the first conjunct means that John intended that very event to be the killing of his uncle; and since that assumption is false, the proposition is, correctly, false.

(46) Kill(John, Uncle) & Intend(John, Kill(John, Uncle))

(47) $\exists e$ [Kill(*e*, John, Uncle) & Intend(John, Kill(*e*, John, Uncle))]

While the representation in 44 and 47 captures correctly the semantics of the Intentional class of adverbs, there is another property of this class that has generated considerable interest, having been discussed by Lakoff [1972], Thomason and Stalnaker [1973], and McConnell-Ginet [1981]: the phenomenon of passive-sensitivity.⁹ When certain semantic conditions apply, it is possible to have two readings for 48 (with the positional variants favoring one reading over the other, but not always excluding the unfavored reading), one corresponding to the situation in which Joan is reluctant and one corresponding to the situation in which Fred is reluctant; these readings are paraphrased in 49 and 50:

(48a) Reluctantly, Fred was taught by Joan.

(48b) Fred reluctantly was taught by Joan.

(48c) Fred was reluctantly taught by Joan.

(48d) Fred was taught reluctantly by Joan.

(48e) Fred was taught by Joan reluctantly.

(49) Joan was reluctant to teach Fred.

(50) Fred was reluctant to be taught by Joan.

The possibility that either the subject or the agent (when the latter is not the subject) is the reluctant participant in the event constitutes the passive-sensitivity of the adverb. The semantic restriction governing the phenomenon of passive-sensitivity is the relevance of the potential of control¹⁰ by the participant over the execution of the action; the adverbs

⁹This term was first used by McConnell-Ginet [1981:145].

¹⁰The potential for control, rather than control itself, is the correct way of stating the condition because adverbs like *unwittingly* or *unwillingly* indicate not that the participant has control over the action, but only that the potential for control was there, yet it was thwarted or not acted upon by virtue of ignorance, deceit, or some outright external force.

for which this is true are not just the Intentional adverbs but the Ability adverbs as well:

- (51) Stupidly, the assistant was caught by the police while she was leaving the mayor's house.
- (52) The assistant was stupid to be caught by the police while she was leaving the mayor's house.
- (53) The police were stupid to catch the assistant while she was leaving the mayor's house.

While this ambiguity is a clear case for the necessity of another argument to the adverb besides the proposition, one still needs to explain how the two readings are possible under the conditions specified above. Superficially, the condition appears to be a disjunctive one: the other argument to the adverb must be either the agent or the subject. In the case of active sentences, agent and subject are the same, so only one reading is possible; in the case of passive sentences, agent and subject are distinct roles in the *surface structure*, so we have the ambiguity. McConnell-Ginet proposes that in the subject reading the adverb is associated with the higher verb, that is, with the passive auxiliary *be*, while in the agent reading the adverb is associated with the lower verb, the passive participle. While this solution is in itself somewhat questionable—the *by*-phrase that contains the agent argument in the passive construction is certainly outside of the VP immediately dominating the passive participle, no matter what one's analysis of auxiliaries may be—when one examines evidence from languages with morphological passives instead of syntactic ones, McConnell-Ginet's analysis is untenable. In such languages, her analysis would predict that there is only one reading, i.e., the agent-oriented reading, since there is no higher verb to attach the adverb to for the subject-oriented reading. However, in at least one language with a morphological passive, Japanese, both readings are possible.¹¹ Japanese has a passive suffix that occurs between the verb root and the tense/aspect marker (cf. 54 and 55):

- (54) John-wa Mary-o osie-ta.
John-SBJ Mary-OBJ teach-PAST
'John taught Mary'

¹¹The following data for Japanese were provided to me by Akira Ishikawa and Mariko Saiki.

- (55) Mary-wa John-ni osie-rare-ta
 Mary-SBJ John-AG teach-PASS-PAST
 'Mary was taught by John'

When one inserts the adverb *husyoobusyooni* 'willingly' into 54, one gets only one reading for the sentence, since the agent and the subject coincide in surface structure; however, inserting it into 55 yields an ambiguous sentence, with the subject-oriented reading preferred when the adverb immediately follows the subject, and the agent-oriented reading preferred when the adverb immediately follows the agent phrase:

- (56) John-wa husyoobusyooni Mary-o osie-ta.
 John-SBJ unwillingly Mary-OBJ teach-PAST.
 'John unwillingly taught Mary.'
- (57) Mary-wa husyoobusyooni John-ni osie-rare-ta.
 Mary-SBJ unwillingly John-AG teach-PASS-PAST
 'Mary unwillingly was taught by John.'
- (58) Mary-wa John-ni husyoobusyooni osie-rare-ta.
 Mary-SBJ John-AG unwillingly teach-PASS-PAST
 'Mary was unwillingly taught by John.'

Thus, the distinct readings in both the English and the Japanese cases are not dependent on the number of verbs in the clause, but instead on some deeper semantic relationship that goes against both the syntax and the morphology. The semantics of adverbs like *reluctantly* in 48-52 require that its first argument be an argument in the proposition that makes up the second argument of the adverb. Let us consider grammatical voice as an operation on logical form which makes available one argument (call it the "subject", reflecting its final surface-syntactic status) over the others, so that the (unmarked) active voice yields $\lambda x.\text{Teach}(e, x, y)$ and the passive alters the form to $\lambda y.\text{Teach}(e, x, y)$. Then, in the agent-oriented reading preferred in 48c-e and paraphrased in 49, the adverb was semantically composed with the predicate before the passive operation was applied, yielding 59, while in the subject-oriented reading preferred in 48a-b and paraphrased in 50, the passive operation was performed before the adverb was composed with the predicate, yielding 60.

- (59) $\exists e[\text{Teach}(e, \text{Joan}, \text{Fred}) \ \& \ \text{Reluctant}(\text{Joan}, \text{Teach}(e, \text{Joan}, \text{Fred}))]$
 (60) $\exists e[\text{Teach}(e, \text{Joan}, \text{Fred}) \ \& \ \text{Reluctant}(\text{Fred}, \text{Teach}(e, \text{Joan}, \text{Fred}))]$

This allows us to reanalyze the condition as a "subject" condition rather than as a disjoint subject-or-agent condition.

However, this means that one reading has to look "inside" the morphological structure of the passive form in order to combine it syntactically with another element of the sentence. This is not a unique and insuperable problem created by our analysis; it is just another example of a fairly widespread phenomenon, the best-known examples of which are given in 61 and 62:

(61) Morphological analysis: [un+[grammatical-ity]]

Semantic analysis: [[un grammatical] ity]

(62) Morphosyntactic analysis: [atomic [scient-ist]]

Semantic analysis: [[atomic scient] ist]

The more closely one analyzes linguistic constructions, the more ubiquitous the mismatches between syntactic structure and logical form turn out to be. For example, the entire analysis of adverbs argued for so far goes partially "against" the syntax of adverbs, with the division between [syntactically] sentential and verbal adverbs being different from the one between operators and predicates. While a rough-hewn correspondence between morphosyntactic structure and the structure of logical form is quite apparent, it is clear that the simple rule-to-rule hypothesis of compositionality it suggests must be refined considerably in order to account for the type of behavior described here.

3.4 Some Arguments for Some Arguments

Having described the different logical forms found in the adjective and adverb classes considered in this paper, it remains to examine the large number of AA's that are predicates and to determine the number and type of arguments the predicates of each class take.

There are three major criteria for establishing the need for an argument to a predicate. The first is that the concept denoted by the predicate necessarily implies the participation in some way of other entities—usually objects and agents, but also events, propositions, and even more exotic entities like the **FACT(P)** forms proposed earlier. The second is that the identity of those entities is not automatically predictable from the information already encoded in the predicate's semantics. The value-as-argument analysis of properties discussed in Section 2.2 did not satisfy this criterion, since in all cases the identity of the attribute is can be predicted from the

semantics of the predicate (the “value”); this was accounted for by determining that attributes are actually higher-level types and do not participate directly in the relation between the so-called “value” and the individual. The third criterion is whether or not the putative argument can actually appear in the utterance as a syntactic constituent dependent on the predicate word. Its presence means that some intimate relation holds between it and the predicate independent of contextual factors and the semantics of the predicate. Let us now examine the adverbial predicates and their adjectival counterparts in order to determine the relationship between them from the standpoint of how many arguments they take, what type they are, and which surface-syntactic form seems to be the basic one and which one, derived.

We have already seen that the agent (or rather, “subject”) argument for Intention and Ability adverbs is a necessary argument of the predicate because it can vary in some circumstances, namely in passive constructions; thus, its identity is not predictable from the adverb’s semantics. The adjective has the same meaning, even though it can be found attributed to an agent without the mention of an event:

- (63) John is clever.
- (64) John is clever at playing the dictionary game.
- (65) John was clever to wait seven years before opening the 1974 Pom-mard.

The reason for this is that 63 is actually ambiguous, depending on the context: one could be uttering it in order to convey the idea expressed, for example, in 64 or 65 when the additional information supplied by the complements of the latter sentences is understood in the context. Out of context, of course, the usual interpretation of 60 would be that John is typically or generally clever in whatever he does—“generically” clever, so to speak (or, to be more specific, the second variable of **Clever(John, x)** is bound by a generalized quantifier **G**, as described by Farkas [1982]). Note that the generic-event reading covers both events and *the fact that S* types: John’s general cleverness covers what he does as well as how he does it. This supports the generalized quantifier binding that the generic reading implies: the domain of the variable is not restricted in any way. Finally, it is obvious that sentences with explicit complements such as 64 and 65 will require a predicate with two arguments for the adjective, which strongly supports treating 63 as taking two arguments as well.

It turns out that, for almost all adverbs that are predicates on events and that have adjective counterparts like *clever* or *willing*, such adjectives are semantically identical to the adverbs. For example, with Behavior adverbs such as *rudely*, the adjective constructions semantically require an event as well as an agent, which is generic if unstated, as in 66, and which can be explicitly mentioned, as in 67 and 68.¹²

(66) Thomas is rude.

(67) Thomas was rude in speaking to the teacher.

(68) Thomas was rude to pull his sister's hair.

The Evaluative adverbs such as *fortunately* and *luckily* also are two-place predicates. In many cases the second argument is left to be implied by contextual factors, but it can appear as a distinct constituent in either the surface adjectival or adverbial form of the predicate:

(69) Fortunately for Tom, he left the house before the slide.

(70) John was lucky to get his application in before the deadline.

Unlike some of the other classes we have described, the second argument to Evaluative class forms may be related very indirectly to the action or state of affairs described in the first argument.

(71) Luckily for George, Harry threw the ball to Fred.

The Emotional State adverbs, on the other hand, seem to be one-place predicates that are syntactically derived from but semantically identical to their adjectival counterparts, which are one-place predicates on individuals, but do not have a different semantic form. The sentential-adverb form that *bitter* takes in 66 does not imply that the emotional state that Mary is in is related directly to the event which forms the main predication of the utterance. In fact, the form in 72 is a historical innovation based on the sentence type found in 73 and 74:

¹²The form in 68, with a *to* + infinitive construction (called here *to* *Vinf*), has only the factive-sentential reading, while the form in 64, with the *in* + gerund construction (called here the *at/in* *Ving* construction, since other variants take *at* instead of *in*), exhibits either the verbal or the factive readings, though the verbal reading is preferred. This distribution is a general fact about these nonfinite constructions: the *at/in* *Ving* constructions are used for verbal readings, the *to* *Vinf* constructions for the factive-sentential readings. The only exception to this rule is the use of a [gapped] *for-to* complement with Facility adverbs (see below).

- (72) Bitterly, Mary left the apartment for the last time.
- (73) Bitter, Mary left the apartment for the last time.
- (74) Mary, bitter, left the apartment for the last time.

Further evidence supporting this argument is that when *bitter* is a predicate adjective, it cannot take a complement:

- (75) Mary was bitter *to leave/*?in leaving the apartment for the last time.

The other one-place predicates are somewhat more complicated in their derivational structure: in some cases, there are actually other arguments, most of which are related to the phenomenon of gradability. The arguments contributed by the semantics of gradability will be discussed briefly at the end of this paper. We are primarily interested, however, in the relationship of the argument structure to the representation of events that has been proposed so far.

The Measure AA's actually have a very complicated semantics when it comes to the number of arguments and the existence of derived forms, although they are all verbal adverbs. Let us begin by considering those AA's that describe properties of processes or events. These include such AA's as *successfully* and *slowly*. Their primary use is as modifiers of events:

- (76) Gerald slowly picked himself up off the floor.
- (77) Marcel successfully merged his company with Limelight Industries.

The adjectival counterparts that are identical in logical form modify action nominalizations, since they are predicates on events:

- (78) The destruction of the city by the Germans was rapid.
- (79) The merger of the two chemical companies was successful.

However, there are adjectival forms of these AA's that take an individual as an argument, rather than an event:

- (80) Muhammed is slow.
- (81) Marcel is successful.

As has been pointed out by Uszkoreit [1980] and others, there is an understood role in 80 or 81 in which Muhammed is slow or Marcel is successful; this can be made explicit, as in 82 or 83:

- (82) Muhammed is slow at learning languages (but fast at programming).
- (83) Marcel is successful in merging companies (but not at composing operas).

It is also possible for 80 and 81 to be interpreted as meaning that, as a rule, Muhammed is slow or Marcel is successful in any activity either might undertake. Even 82 or 83 are generic as well, in that the role expressions are generic.

Nevertheless, in the original or "basic" uses of the AA in reference to an event, there is no need for an additional argument for, say, the subject: success in merging the companies may or may not be attributable to Marcel in 77 (cf. 79, which could be referring to the same event and does not refer to Marcel at all). Examples 80–83, however, indicate that actions of some type associated with the individual about whom the AA is predicated are generally slow, successful, etc. These adjectival uses are secondary applications that are derived from the primary one-place event predicate; they add a second argument and thus have the form $P(r,x)$, meaning roughly "x is P at doing r". The variable r denotes a *role*, that is, a generic activity such as running or learning languages, in which the individual mentioned in the other argument of the predicate is interpreted as the agent.

The distinctions are more complicated when one has an AA like *beautiful(ly)* which, in addition to modifying events, can also directly modify individuals—in this case, describing physical appearance. Thus, to borrow some well-known examples from Siegel [1976], we have the following two sentences and three logical forms, in which *Beautiful'* denotes the two-place predicate derived from *Beautiful*:^{13,14}

- (84) Marya dances beautifully.
- $\exists e[\text{Dance}(e, \text{Marya}) \ \& \ \text{Beautiful}(e)]$

¹³The interpretation of *beautiful dancer* in the first logical form listed under 85 is not a result of the mismatch phenomenon such as in example 62 above. Uezkoreit [1980] pointed out that the role variable in the derived adjectival form does not necessarily refer to the role denoted by the head noun in sentences like 85; his example is *John is a good sophomore* where in the context John is good at playing football. Thus, the role variable in the first logical form listed in 85 could theoretically refer to roles other than dancing.

¹⁴It is possible that the *Beautiful* predicate referring to physical appearance may be distinct (though obviously related) from the one-place predicate that characterizes events; see Footnote 15.

- (85) Marya is a beautiful dancer.
 Dancer(Marya) & $\exists r$ [Beautiful'(r,Marya)] or
 Dancer(Marya) & Beautiful(Marya)

The same sort of argument applies mutatis mutandis to Facility AA's, words expressing the facility of performing an action such as *easy/easily*, and *difficult/with difficulty*—the class of so-called "Tough-Movement" adjectives. They refer to actions as in 86; the adjectival use in 87 implies an action in which the individual of whom the adjective is predicated is a participant, e.g. the actions exemplified in the *to Vinf* complements in 88. The uses in 87 and 88 represent a two-place predicate derived from the one-place event predicate in 86, which does not specify any participant in the action as rendering the action "easy":

- (86) Yolanda easily shot the arrow into the bullseye.
 (87) This exam is difficult.
 (88) This exam is difficult to read/to understand/to pass.

The chief difference between this class and the other Measure AA's is that the individuals of whom derived Facility adjectives are predicated must participate in the relevant actions as direct objects or as other affected participants, whereas the thematic relation between participant and action for the derived Measure adjectives is much freer (though it is usually the agent):

- (89) *Daniel is easy to tease people. [=Daniel teases people easily]
 (90) Daniel is successful at avoiding the draft.
 (91) Daniel was successful in not being picked to head the commission.

Furthermore, the surface syntax for indicating the relevant role in a derived Facility AA form is a (*for-*)*to* complement rather than the *at/in Ving/Vnom* expression used for the derived Measure AA's.¹⁵

¹⁵The two derivational processes appear to be in complementary distribution. There is one antonymic pair of AA's that seems to function both as Measure and as Facility AA's: *good/well* and *bad/badly*; see examples a-c. However, the meaning of a is clearly not related to the meaning of b in the way it is related to the meaning of c:

- (a) John played the Hammerklavier Sonata well.
 (b) The Hammerklavier Sonata is good to play. [e.g. in order to get good reviews]
 (c) John is good at playing Beethoven.

The predicate in b seems to be idiosyncratic and should be treated as distinct from the

Measure AA's actually take one or perhaps two other arguments that are related to their gradability rather than their applicability to both individuals and events. One of the defining characteristic of Measure AA's is that they are gradable—that is, the range of properties of a single attribute range over a [usually unidimensional] continuum, or at least, in the case of subjective measures like *good*, *cute* or *ugly*, are perceived to be ranked in such a way. Many of these AA's (*tall*, *little*, *shallow*, etc.) apply only to individuals and not events, and so do not share in the complications discussed above. In addition, many of the two-place AA's discussed above, such as the Behavior and Ability types (but not the Intentional AA's) are gradable, and so the following remarks pertain to them as well. Since a great deal has been written about gradability, and since gradability is somewhat peripheral to the basic issues surrounding the logical form of adverbs, I will present a brief, simplified discussion of the issues with respect to the predicate-argument structure of AA's.

The first additional argument taken by gradable AA's, whose existence is now relatively uncontested, is the "reference set" argument, denoting the class of individuals from which is derived the "average" value of the gradable property against which the AA in question is to be evaluated. To take a simple spatial-dimension term as an example, not only is *tall* vague as to what degree of height is intended, it is also indeterminate as to the assumable neutral point or region above which someone is considered to be tall and below which someone is to be considered short. Thus, in the following sentences, John and Jim may be the same height, yet one is "tall" and the other "short":

(92) John is tall for a fourteen-year-old.

(93) Jim is short for a professional basketball player.

Reference sets are relevant for "subjective" measure terms and other gradable terms, even though there is no universally agreed-upon metric that can be imposed on the domain:

derivational pair in *a* and *c*. However, if this is true, then it is more difficult to argue that the use of *good* in *d*, in which it is intended to refer to some inherent moral quality of the individual, is the same predicate as the one in *a* rather than another distinct but lexicosemantically related form:

(*d*) Sam is good.

Such a proliferation of semantically related but distinct predicates may lead to difficulties later on.

- (94) Jim is a good dancer, for a wrestler.
- (95) Freddy is awfully rude, even for an eight-year-old.

Since reference sets are essential for the correct semantic interpretation of the AA, are not predictable from other information available in the utterance, and can be introduced into the utterance as independent arguments, one must include them as such. Thus, the use of *good* in an utterance like 96 actually has three arguments—individual, role, and reference set—that can be specified independently. Although, taken out of context, 97 is the usual interpretation of 96, the actual interpretation in a given context may be, e.g. 98:¹⁶

- (96) The violinist is good.
- (97) The violinist is good at playing the violin, for a violinist.
- (98) The violinist is good at leading the musician's union, for a shy and reclusive person.

A *more controversial* question is whether sentences like 99–101 have a fourth argument that refers to the degree to which the individual possesses the value denoted by the AA with respect to the appropriate reference set (and role, in 101):

- (99) Jim is six feet tall (*for a basketball player).
- (100) Jim is pretty/very/extremely tall (for a basketball player).
- (101) The violinist is pretty/very/extremely good.

While the specific value in 99 is most likely an argument—it is implied by the semantics of gradability, it is not predictable, and it can be represented explicitly (albeit optionally) in the utterance—that value term is syntactically parallel to the vaguer terms in 100 and 101, which are called “amplifiers” by Quirk et al. [1972:246, 444–51] and which denote a vague value on the scale represented by the AA. These latter forms have not been considered as arguments in the past, but the evidence from more precise measure phrases such

¹⁶If the adjective is attributive, it is extremely difficult though not impossible to obtain a reading in which the intended role is different from the activity associated with the head noun (usually an agentive nominal form). Moreover, the reference set is also usually interpreted as the same set as the one referred to by the part of the NP that follows the measure term in surface structure. For example, e.g. *a good Baroque violin player* would normally be judged against the reference set of Baroque violin players, not violinists in general or players in general.

as *six feet* suggest that they might be. However, precise measure phrases do not cooccur with phrases indicating the reference set, while amplifiers do; this suggests that, whatever analysis is chosen, more subtle constraints are required.

These brief comments on gradability are, of course, not conclusive. They are intended only to indicate that other factors will contribute to the argument structure of certain adjectives and adverbs, and that such factors must be distinguished from those that are consequences of the interactions of AA's with events (for example, role arguments are derivative arguments from the use of properties of events when predicated of individuals, while reference set arguments are part of the structure of gradability).

4 Conclusion

Our research on the semantics of adverbs and adjectives touches upon several interesting issues of general concern. In particular, it has led to arguments supporting the existence not only of an event variable for actions (but excluding states), but also for the state of affairs concept (represented by the **FACT** operator) as a distinct phenomenon. It has also led to further evidence for the separation of surface syntax from logical form. It is interesting to note, however, that adverbs themselves comprise a relatively unified phenomenon: a small class of operators on the one hand, a variety of predicates on the other. The lexical-semantic concepts denoted by specific adverbs (as represented in Section 2.1) are extremely diverse, belonging to domains such as mental states that have been explored very little until recently. Our analysis has, we hope, clarified a number of issues raised by the logical forms of most adverbs and adjectives, so that further research in these areas can be done on a firmer logical basis than has been possible hitherto.

Bibliography

- Barwise, Jon and John Perry, 1983: *Situations and Attitudes* (MIT Press, Cambridge, Mass.).
- Bellert, Irena, 1977: "On semantic and distributional properties of sentential adverbs", *Linguistic Inquiry*, vol. 8, no. 2, pp. 337-351.
- Bresnan, Joan, 1982: "Polyadicity", *The Mental Representation of Grammatical Relations*, J. Bresnan, ed. (MIT Press, Cambridge, Mass.), pp. 149-172.

- Chisholm, R. M., 1966: "Freedom and action", *Freedom and determinism*, K. Lehrer, ed. (Random House, New York).
- Clark, E. V. and H. H. Clark, 1979: "When nouns surface as verbs", *Language*, vol. 55, no. 4, pp. 767-811.
- Davidson, Donald, 1967: "The logical form of action sentences", *The Logic of Decision and Action*, N. Rescher, ed. (University of Pittsburgh Press, Pittsburgh, Penn.).
- Downing, Pamela, 1979: "On the creation and use of English compound nominals", *Language*, vol. 53, no. 4, pp. 810-842.
- Ernst, Thomas, 1984a: *Towards an Integrated Theory of Adverb Position in English* (Ph.D. dissertation, Indiana University, Bloomington, Indiana). Reprinted by the Indiana University Linguistics Club.
- , 1984b: Manner adverbs and the S/VP Distinction (unpublished manuscript, Indiana University).
- Farkas, Donka, 1982: *Intensionality and Romance Subjunctive Relatives* (Ph.D. dissertation, University of Chicago, Chicago, Illinois). Reprinted by the Indiana University Linguistics Club.
- Hintikka, Jaakko, 1971: "Semantics for propositional attitudes", *Philosophical Logic*, J. W. Davis, ed. (Reidel, Dordrecht).
- Jackendoff, Ray, 1972: *Semantic Interpretation in Generative Grammar* (MIT Press, Cambridge, Mass.).
- Kiparsky, Paul and Carol Kiparsky, 1970: "Fact", *Progress in Linguistics*, M. Bierwisch and K. Heidoiph, ed. (Mouton, The Hague). Reprinted in *Semantics*, Danny Steinberg and Leon Jakobovits, ed. (Cambridge University Press, Cambridge, 1971).
- Lakoff, George, 1972: "Linguistics and natural logic", *Semantics of Natural Language*, D. Davidson and G. Harman, ed. (Reidel, Dordrecht).
- Levi, Judith, 1978: *The Syntaz and Semantics of Complex Nominals* (Academic Press, New York).
- McConnell-Ginet, Sally, 1981: "Adverbs and logical form", *Language*, vol. 58, no. 1, pp. 144-184.
- Moore, Robert, 1981: "Problems in logical form", *Proceedings of the 19th Annual Meeting of the Association for Computational Linguistics*, Stanford, California.
- Nunberg, Geoff, 1979: "The nonuniqueness of semantic solutions: polysemy", *Linguistics and Philosophy*, vol. 3, no. 2, pp. 143-184.
- Quirk, Randolph, S. Greenbaum, G. Leech, and J. Svartvik, 1972: *A Grammar of Contemporary English* (Longmans, London).

- Searle, John, 1980: "The intentionality of intention and action", *Cognitive Science*, vol. 4, pp. 47-70.
- Siegel, Muffy, 1976: *Capturing the Adjective* (Ph.D. dissertation, University of Massachusetts, Amherst, Massachusetts).
- Thomason, Richmond and Richard Stalnaker, 1973: "A semantic theory of adverbs", *Linguistic Inquiry*, vol. 4, no. 2, pp. 195-220.
- Uszkoreit, Hans, 1980: "Do English adjectives come in two types?" (unpublished manuscript, SRI International).
- Woods, William, 1975: "What's in a link: foundations for semantic networks", *Representation and Understanding: Studies in Cognitive Science*, D. Bobrow and A. Collins, ed. (Academic Press, New York).

Appendix D

BELIEF AND INCOMPLETENESS

SRI International



Belief and Incompleteness

Technical Note 319

July 13, 1984

Kurt Konolige
Computer Scientist

Artificial Intelligence Center
Computer Science and Technology Division

To appear in *Formal Theories of the Common-Sense World*,
edited by Jerry Hobbs.

This research was made possible in part by a gift from the System Development Foundation. It was also supported in part by Contract N00014-80-C-0296 from the Office of Naval Research, and by Contract F49620-82-K-0031 from the Air Force Office of Scientific Research.

333 Ravenswood Ave. • Menlo Park, CA 94025
415 326-6200 • TWX 910 3-2046 • Telex 334 486

Contents

1. INTRODUCTION	1
2. TWO PROBLEMS IN THE REPRESENTATION OF BELIEF	5
3. THE DEDUCTION MODEL OF BELIEF	11
3.1 Planning and Beliefs: the Belief Subsystem Abstraction	11
3.2 A Formal Model of Belief	14
3.3 Properties of Deduction Structures	16
4. THE LOGIC FAMILY B	25
4.1 Block Tableaux	25
4.2 The Language of B	30
4.3 A Sequent System for B	32
4.4 The Nonintrospective Logic Family BK	35
5. THE PROBLEMS REVISITED	43
6. OTHER FORMAL APPROACHES TO BELIEF	53
6.1 The Possible-Worlds Model	53
6.2 Syntactic Logics for Belief	58
7. CONCLUSION	61
ACKNOWLEDGEMENTS	63
REFERENCES	65

1. Introduction

Two artificially intelligent (AI) computer agents begin to play a game of chess, and the following conversation ensues:

S₁: Do you know the rules of chess?

S₂: Yes.

S₁: Then you know whether White has a forced initial win or not.

S₂: Upon reflection, I realize that I must.

S₁: Then there is no reason to play.

S₂: No.

Both agents are state-of-the-art constructions, incorporating the latest AI research in chess playing, natural-language understanding, planning, etc. But because of the overwhelming combinatorics of chess, neither they nor the fastest foreseeable computers would be able to search the entire game tree to find out whether White has a forced win. Why then do they come to such an odd conclusion about their own knowledge of the game?

The chess scenario is an anecdotal example of the way inaccurate cognitive models can lead to behavior that is less than intelligent in artificial agents. In this case, the agents' model of belief is not correct. They make the assumption that an agent actually knows all the consequences of his beliefs. *S₁* knows that chess is a finite game, and thus reasons that, in principle, knowing the rules of chess is all that is required to figure out whether White has a forced initial win. After learning that *S₂* does indeed know the rules of chess, he comes to the erroneous conclusion that *S₂* also knows this particular consequence of the rules. And *S₂* himself, reflecting on his own knowledge in the same manner, arrives at the same conclusion, even though in actual fact he could never carry out the computations necessary to demonstrate it.

We call the assumption that an agent knows all logical consequences of his beliefs *consequential closure*. This assumption is clearly not warranted for either mechanical or human agents, because some consequences, although they are logically correct, may not be computationally feasible to derive. This is in fact illustrated by the chess scenario. Unfortunately, the best current formal models of belief on which AI systems are based have a *possible-worlds* semantics, and one of the inherent properties of these models is consequential closure. While such models are good at predicting what consequences an agent could *possibly* derive from his beliefs, they are not capable of predicting what an agent *actually* believes, given that the agent may have resource limitations impeding the derivation of the consequences of his beliefs.

The chess scenario illustrates one source of logical incompleteness in belief derivation, namely, an agent may not have enough computational resources to actually derive some result. We will identify several others in Section 2, by presenting a problem in belief representation that we have called the Not-So-Wise-Man Problem, a variation of the familiar Wise Man Puzzle. Not surprisingly, this problem involves reasoning about beliefs an agent does *not* have, even though they are logical consequences of his beliefs. The representational problems posed by the chess scenario and the not-so-wise-man problem cannot be solved within the framework of any model of belief that assumes consequential closure.

In this paper we introduce a new formal model of belief, called the *deduction model*, for representing situations in which belief derivation is logically incomplete. Its main feature is that it is a symbol-processing model: beliefs are taken to be expressions in some internal or "mental" language, and an agent reasons about his beliefs by manipulating these syntactic objects. Because the derivation of consequences of beliefs is represented explicitly as a syntactic process in this model, it is possible to take into account the fact that agents can derive some of the logically possible consequences, but in many cases not all of them. When the process of belief derivation is logically incomplete, the deduction model does not have the property of consequential closure.

Symbol-processing models of belief in themselves are not new (see, for example, Fodor [10], Lycan [23], and Moore and Hendrix [31] for some philosophical underpinnings, and McCarthy [26], Perlis [33], and Konolige [19] for AI approaches). The deduction model

presented here differs significantly from previous approaches, however, in two respects. First, it is a formal model: beliefs are represented in a mathematical framework called a *deduction structure*. The properties of the deduction model can be examined with some preciseness, and we do so in Section 3. Second, we have found sound and complete logics for the deduction model. One of these, B , is presented in Section 4, and used in the solution of the problems in Section 5. An important property of the deductive belief logic B is that it can serve as a basis for building computer agents that reason about belief. We have been able to find a number of interesting proof methods for B that have reasonable computational properties. Although the exposition of these methods is beyond the scope of this paper, at the appropriate points we will show how the design of the logic was influenced by computational considerations.

The nature of the deduction model and its logic B is further analyzed by comparing B to modal logics based on a possible-worlds semantics in Section 6. An important result is that the deduction model exhibits a correspondence property: in the limit of logically complete deduction, B reduces to a modal logic with possible-worlds semantics. Thus the deduction model dominates the possible-worlds model, while retracting the assumption of consequential closure.

The material for this paper was abstracted from the author's dissertation work (Konolige [21]). Because of the limited scope of this paper, we are not able to do more than mention in passing several interesting topics that are a part of the deduction model and its logics. Among these are efficient proof methods, the formal semantics and completeness proofs, extensions to B that permit quantifying-in, and introspection properties (beliefs about one's own beliefs). Interested readers can consult the dissertation itself for a fuller exposition.

2. Two Problems in the Representation of Belief

In this section we introduce three ways in which an agent may be incomplete in reasoning from his beliefs: *resource-limited incompleteness*, *fundamental logical incompleteness*, and *relevance incompleteness*. We argue that it is important for AI systems that reason about belief to be able to represent each of these, and offer two anecdotal problems to support this contention.

THE CHESS PROBLEM. *Suppose an agent knows the rules of chess. It does not necessarily follow that he knows whether White has a winning strategy or not.*

The chess problem, on the face of it, seems hardly to be a representational problem at all. Certainly its statement is true: no agent, human or otherwise, can possibly follow out all the myriad lines of chess play allowed by the rules to determine whether White has a strategy that will always win. What kind of model of belief would lead us to expect an agent to know whether White has a winning strategy? As we stated in the introduction, any model that does not take resource limitations into account in representing an agent's reasoning about the consequences of his beliefs has this behavior. Within such a model, we could establish the following line of argument.

Chess is a finite game,¹ and so it is possible, in theory, to construct a complete, finite game tree for chess, given the rules of the game. The question of White's having a winning strategy is a property of this finite game tree. If for every counter Black makes, White has a move that will lead to a win, then White has a winning strategy. Thus, White's having a winning strategy is a consequence of the rules of

¹ The finiteness of chess is assured by the rule that, if 50 moves occur without a pawn advance or piece capture, the game is a draw.

chess that can be derived in a finite number of simple steps. If an agent believes all the logical consequences of his beliefs, then an agent who knows the rules of chess will, by the reasoning just given, also know whether White has a winning strategy or not.

The chess problem is thus a problem in representing reasoning about beliefs in the face of resource limitations. The inference steps themselves are almost trivial; it is a simple matter to show that a move is legal, and hence to construct any position that follows a legal move from a given position. But while the individual inferences are easy, the number of them required to figure out whether White has a forced win is astronomical and beyond the computational abilities of any agent. We call this behavior *resource-limited incompleteness*. A suitable model of belief must be able to represent situations in which an agent possesses the inferential capability to derive some consequence of his beliefs, but simply does not have the computational resources to do so.

THE NOT-SO-WISE-MAN PROBLEM. *A king, wishing to know which of his three advisors is the wisest, paints a white dot on each of their foreheads, tells them there is at least one white dot, and asks them to tell him the color of their own spots. After a while the first replies that he doesn't know; the second, on hearing this, also says he doesn't know. The third then responds, "I also don't know the color of my spot; but if the second of us were wiser, I would know it."*

The not-so-wise-man problem is a variation of the classic Wise Man Puzzle, which McCarthy (in [24] and [25]) has used extensively as a test of models of knowledge. In the classic version, the third wise man figures out from the replies of the other two that his spot must be white. The "puzzle" part is to generate the reasoning employed by the third wise man. The reasoning involved is really quite complex and hinges on the ability of the wise men to reason about one another's beliefs. To convince themselves of this, readers who have never tried before may be interested in attempting to solve it before reading the solution below.

Solution to the Wise Man Puzzle: *the third wise man reasons: "Suppose my spot were black. Then the second of us would know that his own spot was white, since he would know that, if it were black, the first of us would have seen two black spots and would have known his own spot's color. Since both answered that they had no knowledge of their own spot's color, my spot must be white."*

The difficulty behind this puzzle seems to lie in the nature of the third wise man's reasoning about the first two's beliefs. Not only must he pose a hypothetical situation (*Suppose my spot were black*), but he must then reason within that situation about what conclusions the second wise man would come to after hearing the first wise man's response. This in turn means that he must reason about the second wise man's reasoning about the first wise man's beliefs, as revealed by his reply to the king. Reasoning about beliefs about beliefs about beliefs... we call reasoning about *iterated* or *nested beliefs*. It can quickly become confusing, especially when there are conditions present concerning what an agent does *not* believe.

In the Wise Man Puzzle, nested belief contributes to the complexity of the reasoning involved. The third wise man must reason about what the second wise man does not know (the color of his own spot); in doing this, he must also reason about the second wise man's reasoning about what the third wise man does not know (the color of *his* own spot). It is particularly annoying and troublesome to keep track of who believes what after several occurrences of not-believing in a statement of nested belief. Because human agents find it so difficult, the Wise Man Puzzle is thought to be a good test of the *competence* of any model of belief. If one can state the solution to the puzzle within the framework of Model X, so the reasoning goes, then Model X is at least good enough to show what might conceivably be concluded by agents in complicated situations involving nested beliefs.

It is possible to solve the Wise Man Puzzle within the confines of belief models that assume consequential closure (see, e.g., McCarthy [24], [25] or Sato [38]). Such models make the assumption that every agent believes other agents' beliefs are closed under logical consequence, and so on to arbitrary depths of belief nesting. While this is an accurate assumption if one is trying to model the competence of ideal agents (which is what the Wise Man Puzzle seeks to verify), it cannot represent interesting ways in which reasoning about complicated nested beliefs might fail for a less-than-ideal agent. This is the import

of the not-so-wise-man problem. From the reply of the third wise man, it appears that the second wise man lacks the ability to deduce all the consequences of his beliefs. The representational problem posed is to devise interesting ways in which the second wise man fails to be an ideal agent, and then show how the third wise man can represent this failure and reply as he does.

The not-so-wise-man problem does not seem to fall into the category of resource-limited incompleteness mentioned in the chess problem, since the computational requirements of the inferences are not particularly acute. We can identify at least two other types of incompleteness (there may certainly be more) that are interesting here and would be useful to represent. In one of these, the second wise man may have incomplete inferential procedures for reasoning about the other wise men's beliefs, especially if tricky combinations of *not-believing* are present. Suppose, for instance, the second wise man were to see a black spot on the third wise man, and a white spot on the first wise man (this is the hypothetical situation set up by the third wise man in solving the classic puzzle). If he were an ideal agent, he would conclude from the first wise man's reply that his own spot must be white (by reasoning: *if mine were not white, the first of us would have seen two black spots and so claimed his own as white*). But he may fail to do this because his rules for reasoning about the beliefs of the first wise man simply are not powerful enough. For example, he might never consider the strategy of assuming that his spot was black, and then asking himself what the first wise man would have said. In this case, the second wise man's inferential process, even when given adequate resources, is just not powerful enough in terms of its ability to arrive at simple logical conclusions. To apply an analogy from high-school algebra: a student who is confronted with the equation $x + a = b$ and asked to solve for x won't be able to do so if he doesn't know the rule that subtracting equals from each side leaves the equation valid. It is not that the student lacks sufficient mental resources of time or memory to solve this problem; rather, his rules of inference for dealing with equational theories are logically incomplete. To contrast this type of incompleteness with the resource-limited incompleteness described in the chess problem, we call it *fundamental logical incompleteness*.

Another way in which the not-so-wise-man might fail to draw conclusions is if he were to make an erroneous decision as to what information might be relevant to solving

his problem. Although the Wise Man Puzzle has a fairly abstract setting, it is reasonable to suppose that actual agents confronted with this problem would have a fair number of extraneous beliefs that they would exclude from consideration. For example, the not-so-wise-man might be privy to the castle rumor mill, and therefore believe that the first wise man was scheming to marry the king's daughter. A very large number of beliefs of this sort have no bearing on the problem at hand, but would tend to use up valuable mental resources if they were given any serious consideration. One can imagine an *unsure agent* who could never come to any negative conclusions at all, because he would keep on considering more and more possibilities for solving a problem. This agent's reasoning might proceed as follows: *I can't tell the color of my spot by looking at the other wise men. But maybe there's a mirror that shows my face. No, there's no mirror. But maybe my brother wrote the color on a slip of paper and handed it to me. No, there's no slip of paper, and my brother's in Babylon. But maybe ...*

McCarthy (in [27]) first called attention to the problem of representing what is *not* the case in solving puzzles. In the Missionaries and Cannibals Puzzle, why can't the missionaries simply use the bridge downstream to get across? A straightforward logical presentation of the puzzle doesn't explicitly exclude the existence of such a bridge. And, if it did, we could always come up with other modes of transportation that had not been considered beforehand and explicitly excluded. McCarthy called the general problem of specifying what conditions do not hold in a puzzle the *circumscription problem*. By analogy, we call the problem of specifying what beliefs an agent does not have, or does not use in solving a given task, the problem of *circumscriptive ignorance* (see Konolige [20]). Without a solution to this representational problem, all agents will be modeled as unsure agents - never able to reach a conclusion about what they don't believe, even though it is obvious when the set of relevant beliefs is circumscribed.

Of course, if an agent can circumscribe his beliefs, it is possible that he will choose the wrong set of beliefs, and exclude some that actually are relevant. The not-so-wise-man may decide that the beliefs of the first wise man are not germane to the problem of figuring out his own spot's color. Thus, even though he has all the relevant information, and even sufficiently powerful inference rules and adequate resources, he may fail to come to a correct conclusion because he has circumscribed his beliefs in the wrong way. We call this type of incompleteness *relevance incompleteness*.

AD-A162 389

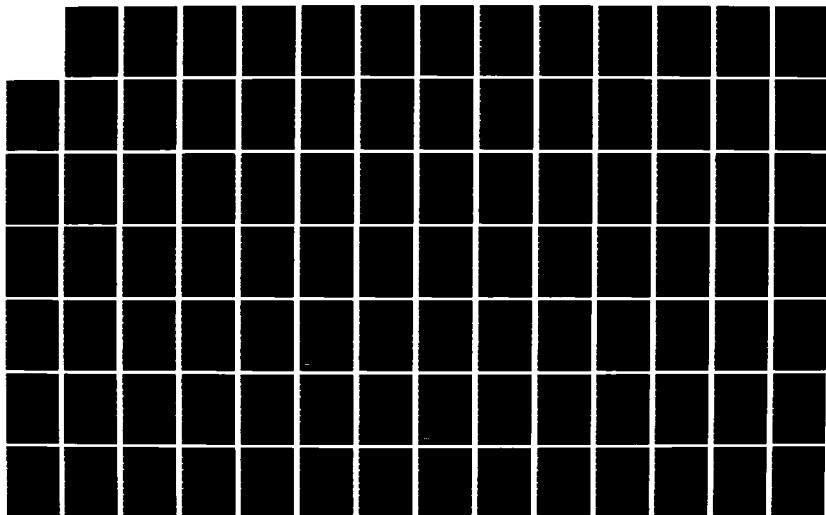
KNOWLEDGE REPRESENTATION AND NATURAL-LANGUAGE SEMANTICS
(U) SRI INTERNATIONAL MENLO PARK CA R C MOORE AUG 85
AFOSR-TR-85-1098 F49620-82-K-0031

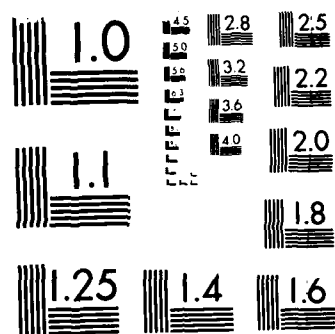
3/4

UNCLASSIFIED

F/G 5/7

NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

Within a model of belief that assumes consequential closure, it is possible to represent circumscriptive ignorance, but only in a relatively limited fashion. If consequential closure is assumed, one can state that an agent is ignorant of some fact which is not a logical consequence of his beliefs (McCarthy [25] uses this technique in his solution to the Wise Man Puzzle). But this clearly does not capture the complete conditions of circumscriptive ignorance, since agents are often ignorant of some of the logical consequences of their beliefs, as in the chess scenario.

Modeling relevance incompleteness (or having the third wise man do so) is impossible if it is assumed that the beliefs of agents are consequentially complete. One simply cannot partition the set of beliefs into those that are either relevant or not to a given problem; *all* the consequences of beliefs are believed. If we try to state the conditions of relevance incompleteness within such a model, we can arrive at a contradiction, where a proposition is both believed (because of the assumption of consequential closure) and not believed (because of the condition of relevance incompleteness).

3. The Deduction Model of Belief

The two belief representation problems can be solved within the framework of a formal model of belief that we call the deduction model. In this section we define the model; in the next we introduce a logic family B as its axiomatization.

The strategy we pursue is to first examine the way typical AI robot planning systems (STRIPS [9], NOAH [37], WARPLAN [42], KAMP [1], etc.) represent and reason about the world. This leads to the identification of an abstract *belief subsystem* as the internal structure responsible for the beliefs of these agents. The characteristics of belief subsystems can be summarized briefly as follows.

1. A belief subsystem contains a list of sentences in some internal ("mental") language, the *base beliefs*.
2. Agents can infer consequences of their beliefs by syntactic manipulation of the sentences of the belief subsystem.
3. The derivation of consequences of beliefs is logically incomplete, because of limitations of the inferential process.

Having identified a belief subsystem as that part of an agent responsible for beliefs, our next task is to define a formal mathematical structure that models it accurately. The decisions to be made here involve particular choices for modeling the various components of a belief subsystem: What does the internal language look like? What kind of inference process derives consequences of the base beliefs? and so on. The formal mathematical object we construct according to these criteria is called a *deduction structure*. Its main components are a set of sentences in some logical language (corresponding to the base beliefs of a belief subsystem) and a set of deduction rules (corresponding to the belief inference rules) that may be logically incomplete. Because we choose to model belief subsystems

in terms of logical (but perhaps incomplete) deduction, we call it the *deduction model of belief*.

3.1. Planning and Beliefs: the Belief Subsystem Abstraction

A robot planning system, such as STRIPS, must represent knowledge about the world in order to plan actions that affect the world. Of course it is not possible to represent all the complexity of the real world, so the planning system uses some abstraction of properties of the real world that are important for its task: e.g., it might assume that there are objects that can be stacked in simple ways (the *blocks world* domain). The state of the abstract world at any particular point in time has been called a *situation* in the AI literature.

In general, the planning system will have only incomplete knowledge of a situation. For instance, if it is equipped with visual sensors, it may be able to see only some of the objects in the world. What this means is that the system has to be able to represent and reason about partial descriptions of situations. The process of deriving beliefs is a *symbol-manipulating or syntactic operation* that takes as input sentences of the formal language, and produces new sentences as output. Let us call any new sentences derived by inferences the *inferable sentences*, and the process of deriving them *belief inference*.

It is helpful to view the representation and deduction of facts about the world as a separate subsystem within the planning system; we call it the *belief subsystem*. In its simplest, most abstract form, the belief subsystem comprises a list of sentences about a situation, together with a process for deriving their consequences. It is integrated with other processes in the planning system, especially the *plan derivation process* that searches for sequences of actions to achieve a given goal. In a highly schematic form, Figure 1 sketches the belief subsystem and its interaction with other processes of the planning system. The belief system is composed of the base beliefs, together with the belief inference process. Belief inference itself can be decomposed into a set of inference rules and a control strategy

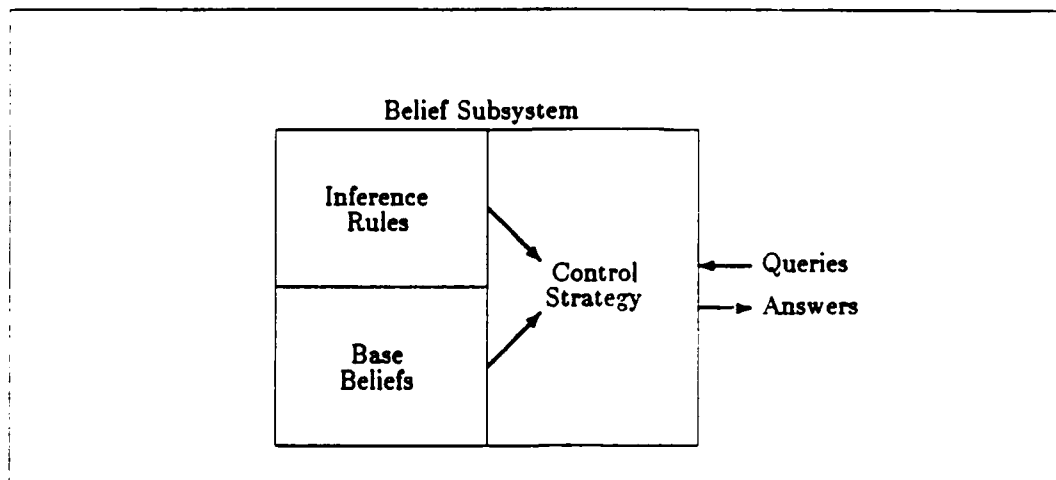


Figure 1. Schematic of a Belief Subsystem

that determines how the rules are to be applied and where their outputs go when requests are made to the belief subsystem.

A belief subsystem defines an agent's beliefs by the action of the inference rules on the base beliefs, under the guidance of the control strategy. Some, but not necessarily all, of the inferable sentences will be beliefs of the agent; which inferable sentences are actually beliefs depends on the details of the control strategy and the resources available for belief inference.

There are two types of requests that result in some action in the belief subsystem. A process may request the subsystem to add or delete sentences in its base beliefs; this happens, for example, when the plan derivation process decides which sentences hold in a new situation. The problem of updating and revising beliefs in the face of new information is a complicated research issue in its own right, and we do not address it here (see Doyle [7] for some related AI research). The second type of request is a query as to whether a sentence is a belief or not. This query causes the control strategy to try to infer, using its rules, that the sentence follows from the base beliefs. It is this process of *belief querying* that we model in this paper.

The above description of the operation of a belief subsystem is meant to convey the idea that in most formal planning systems there is a tight interaction between belief

subsystems and planning. Different systems may deviate from the described pattern to a greater or lesser extent. In some systems, the representation of facts may be so limited, and that of actions so explicit, as to almost obviate the need for belief deduction *per se* (as in some versions of STRIPS). In others, deduction may be used to calculate all the effects of an action by expanding the representation to include situations as objects (as in WARPLAN). Here it is hard to make a clean separation between deductions performed for the purpose of deriving consequences of beliefs and those that establish the initial set of facts about a new situation. However, it is still conceptually useful to regard the belief subsystem as a separate structure and belief derivation as a separate process within the planning system.

3.2. A Formal Model of Belief

The formal mathematical object we use to model belief subsystems is called a *deduction structure*. A deduction structure is a tuple consisting of two sets and will be written as (B, \mathcal{R}) . The set B is a set of sentences in some language L ; It corresponds to the base beliefs of a belief subsystem and its members are referred to as the *base sentences* of the deduction structure. \mathcal{R} is a set of deduction rules for L ; these correspond to the inference rules of a belief subsystem. We demand that deduction structures satisfy the following four conditions.

- Language Property.* The language of a deduction structure is a logical language.
- Deduction Property.* The rules of a deduction structure are logical deduction rules. These rules are sound, effectively computable, and have bounded input.
- Closure Property.* The *belief set* of a deduction structure is the least set that includes the base sentences and is closed under derivations by the deduction rules.
- Recursion Property.* The intended model of deduction structure sentences involving belief is the belief set of another deduction structure.

We discuss each of these properties briefly below. For the interested reader, a more thorough treatment of the mathematical properties of deduction structures is given in the next subsection.

About the only condition we require of L is that it be a *logical language*. Logical languages are distinguished by having a constructable set of syntactic objects, the *sentences* of the language, together with an *interpretation method* (a means of assigning true or false to every sentence with respect to a given state of affairs).

\mathcal{R} is a set of deduction rules that operate on sentences of L . We will leave unspecified the exact form of the deduction rules \mathcal{R} , but we do insist that they operate in the normal manner of deduction rules in some proof-theoretic framework. This means that there is the concept of a *derivation* of a sentence, which is a structure built from effective applications of the rules \mathcal{R} . If p is derivable from the set of sentences Γ in this manner, we write $\Gamma \vdash_{\mathcal{R}} p$, where $\vdash_{\mathcal{R}}$ is a derivation operator for the rules \mathcal{R} . For example, in terms of Hilbert systems (as defined in Kleene [18]), \mathcal{R} would be a set of logical axioms (zero-premise rules) together with *modus ponens* (a two-premise rule). A sentence p would be derivable from the premise sentences $B = \{b_1, b_2, \dots\}$ if there were a Hilbert proof of $(b_1 \wedge b_2 \wedge \dots) \supset p$, using the logical axioms and *modus ponens*.

A deduction structure models beliefs by its *belief set*, which we define as follows.

DEFINITION 3.1.

$$\text{bel}((B, \mathcal{R})) =_{\text{df}} \{p \mid B \vdash_{\mathcal{R}} p\} \quad .$$

The belief set is composed of all sentences that are derivable from the base set B with the rules \mathcal{R} . The derivation operator $\vdash_{\mathcal{R}}$ thus corresponds to the belief inference process of belief subsystems.

For several technical reasons, we restrict the derivation operators allowed in deduction structures to those that satisfy a deductive closure condition. One consequence of this assumption is that the belief set itself obeys a closure property: if the sentence p can be derived from the sentences in a belief set, then it too must be present in the belief set. By making the assumption of deductive closure, the task of formalizing and reasoning about deduction structures is greatly simplified.

It is important to note that deductive closure does not entail *consequential* closure for belief derivation: a set of sentences closed under logically incomplete deduction rules need not contain all logical consequences of the set. This is an important property of

deduction structures, and it enables them to capture the behavior of belief subsystems with resource-bounded control strategies.

Finally, we single out certain sentences of the deduction structure for special treatment, namely the ones that themselves refer to the beliefs of agents. In discussing the not-so-wise-man problem in the previous section, we mentioned that one of the key tests of a belief model is its ability to handle nested beliefs by assuming that agents use the model in representing other agents' beliefs; a belief model that has this characteristic is said to have the recursion property. In terms of deduction structures, the recursion property implies that the sentences of the internal language L that are about beliefs should have another deduction structure as their intended interpretation.

3.3. Properties of Deduction Structures

In this subsection we treat the mathematical properties of deduction structures in some detail, taking care to show how they can model the behavior of belief subsystems of formal AI planning systems.

Language Property

One restriction we place on the language of deduction structures is that sentences of the language have a well-defined (*i.e.*, truth-theoretic) semantics. Such a requirement seems absolutely necessary if we are going to talk about the beliefs of an agent being *true* of the actual world, or, as we will want to do in discussing the rationality of agents, judge the *soundness* of belief deduction rules. Such concepts make no sense in the absence of an interpretation method – a systematic way of assigning meanings to the constructions of the language. As Moore and Hendrix ([31], parts IV and especially V) note, the interpretation method is not something the agent carries around in his head; a belief subsystem is just a collection of sentences, and computational processes manipulate the sentences themselves, not their meanings. One simply cannot put the referent of "Cicero" into a robot's computation device, even if he (Cicero, of course) were alive. But the attribution of semantics to sentences is necessary if an outside observer is to analyze the nature of an agent's beliefs.

How well do actual robot belief subsystems fit in with the assumption of a logical language of belief? AI systems use a variety of representational technologies; chief among these are frames, scripts, semantic nets, and the many refinements of first-order logic (FOL), including PROLOG and the "procedurally oriented" logics of μ -PLANNER, CONNIVER, QA4, and the like. The representations that fall into the latter category inherit their semantics from FOL, despite many differences in the syntactic form of their expressions. But what can we say about the first three? In surface form they certainly do not look anything like conventional mathematical logics; furthermore, their designers often have not provided anything but an informal idea of what the meanings of expressions in the language are. When, after all, is a pair of nodes connected by a directed arc *true* of the world? As Hayes [11] has forcefully argued, the lack of a formal semantics is a big drawback for these languages. Fortunately, on further examination it is often possible to provide such a semantics, usually by transliterating the representation into a first-order language (see Woods [44] and Schubert [39] for a reconstruction of semantic nets in FOL terms, and Brachman [4] for a similar analysis of frames).

In discussing human belief, several philosophers of mind have argued that internal representations that count as beliefs must have a truth-value semantics (see Fodor [10], Field [8], and Moore and Hendrix [31] for a discussion of the many intricate arguments on this subject, especially pp. 48ff. of Field and part V of Moore and Hendrix). However, there almost certainly is a lot more to human belief than can be handled adequately within the framework of a logical language. For example, the question of membership in the belief set of a deduction structure is strictly two-valued: a sentence is either a member of the belief set of a deduction structure, or it is not. If it is, then the assumed interpretation is that the agent believes that sentence to be true of the world. Deduction structures thus do not support the notion of uncertain beliefs directly, as they might do if fuzzy or uncertain membership in the belief set were an inherent part of their structure.¹

One further requirement is that L contain expressions referring to the beliefs of agents. Generally we will take this to be a belief operator whose argument is an expression in L .

¹ However, uncertain beliefs could always be introduced into deduction structures in an indirect manner by letting L contain statements about uncertainty, e.g., statements of the form *P is true with probability 1/2*.

Finally, it is often the case that we will want to freeze the language of deduction structures in order to study their properties at a finer level of detail, e.g., when looking at the behavior of nested beliefs in general or when giving the particulars of the solution to a representational problem. It is convenient to think of the language as being a *parameter* of the formal model. For every logical language L , there is a class of deduction structures $D(L, \rho)$ whose base sets are sentences of the language L (the parameter ρ will be explained in discussing the recursion property below).

Deduction Property

Rules for deduction structures are rules of inference with the following restrictions:

1. The rule is an effectively computable function of sentences of L .
2. The number of input sentences is boundedly finite.
3. The conclusion is sound with respect to the semantics of L .

These restrictions are those normally associated with deduction rules for classical logic, although, strictly speaking, deduction rules need not be sound, if one is just interested in proof-theoretic properties of a logic without regards to semantics.

The fact that belief deduction rules are effectively computable functions means that they can be very complicated indeed. Mathematical logicians are interested in logics with simple deduction rules (such as Hilbert systems) because it is easy to analyze the proof-theoretic structure of such systems. However, for the purpose of deriving proof methods for commonsense reasoning in AI, it is often better to sacrifice simplicity for computational efficiency. For example, Robinson's *resolution rule* [36], which employs a matching process called unification, is a complicated rule that has been widely employed in AI theorem-proving. Another important technique is Weyhrauch's *semantic attachment* [43], a general framework for viewing the results of computation as deductions. In this paper, we will exploit complicated rules that perform deductions that are relatively "large" with respect to the grain size of the predicates, particularly in solving the chess problem of Section 2. Although these "large" deductions could be broken down into smaller steps, it is computationally and conceptually easier to view them as single deductions.

We call an inference rule *provincial* if the number of its input sentences is boundedly finite; deduction rules are always provincial. We thus do not allow inferences about beliefs

that take an infinite number of premises. For example, the following rule of Carnap's is not a valid rule of belief deduction: *if for every individual a : $F(a)$ is a theorem, then $\forall x.F(x)$ is a theorem.*¹ Provincial inference rules have the following interesting property: if α is a consequence of a set of sentences S by the rule, then it is also a consequence of any larger set $S' \supset S$. To see that this must be so, consider that, if α can be derived by the application of provincial rules on the set of sentences S , and S' contains S , then the same derivation can be performed by using S' . Rules that adhere to this property are called *monotonic*. Technically, monotonicity is convenient because it means we can reason about what an agent believes on the basis of partial knowledge about his beliefs. A derivation made using a subset of his beliefs cannot be retracted in the face of further information about his beliefs.

Several types of nonmonotonic (and unsound) reasoning have been of interest to the AI community, specifically

- | | |
|--------------------------|---|
| Belief revision: | the beliefs of an agent are updated to be consistent with new information (e.g., Doyle [7]). |
| Default reasoning: | an agent "jumps to a conclusion" about the way the world is (e.g., McCarthy [27], Reiter [35]). |
| Autoepistemic reasoning: | an agent comes to a conclusion about the world based on his knowledge of his own beliefs (e.g., Collins <i>et al.</i> [6], Moore [30]). |

We are explicitly not trying to arrive at a theory of these forms of reasoning. Indeed, it is helpful here to make the distinction that Israel (in [16]) advocates between inference or reasoning in general (which may have nonmonotonic properties) and the straightforward deduction of logical consequences from a set of initial beliefs. It is the latter concept only that is treated in this paper.

If we wish to accommodate some nonmonotonic theory formally within the framework of the deduction model, then we can view its inferences as deduction rules operating on deduction structure theories as a syntactic whole. McCarthy [27] exploits this approach to formalize a certain type of useful default inference, which he calls *circumscription* (see the description of the not-so-wise-man problem in Section 2). In defining the logic B, we

¹ I am indebted to David Israel for pointing out this example.

will show how to formalize *circumscriptive ignorance*, a type of nonmonotonic inference, in this manner.

Deduction rules for belief subsystems must also be sound. A sound deduction rule is one for which, if the premises are true in an interpretation, then the conclusion will be also (see Kleene [18]). Informally, one would say that sound deduction rules never deduce false conclusions from true premises. *Modus ponens* is an example of such a rule: if p and $p \supset q$ are true, then q must also be.

Soundness of inference is an important property for robot agents in deriving consequences of their beliefs. We would not want a robot who believed the two sentences

- (3.1) *All men are mortal.*
 Socrates is a man.

to then deduce (and hence believe) the sentence

- (3.2) *Socrates is not mortal.*

Soundness is not a critical assumption for the deduction model, since none of the major technical results depend on it. In some cases we may wish to relax it, for example, in modeling the behavior of human syllogistic reasoning, which is often unsound (see Johnson-Laird [17]).

To sum up: deduction structures are restricted to using inference rules which are provincial, sound, and effectively computable. Several interesting types of reasoning, such as reasoning about defaults or one's own beliefs, cannot be modeled directly as deduction rules over sentences. However, they can be incorporated into the deduction model if the input to the rules is taken to be the deduction structure as a whole.

Closure Property

The closure property states that the belief set of a deduction structure is *closed under derivations*. Formally, this amounts to the following conditions on the belief set.

1. $B \subseteq \text{bel}(\langle B, \mathcal{R} \rangle)$.
2. If $\Gamma \subseteq \text{bel}(\langle B, \mathcal{R} \rangle)$ and $\Gamma \vdash_{\mathcal{R}} p$, then $p \in \text{bel}(\langle B, \mathcal{R} \rangle)$.

Since we have defined the belief set in terms of the belief derivation operator \mathbb{B} (Definition 3.1), we can reexpress these as conditions on belief derivation.

(Reflexivity) $\alpha \mathbb{B}_{\rho(i)} \alpha$.

(Closure) If $\Gamma \mathbb{B}_{\rho(i)} \beta$ and $\beta, \Sigma \mathbb{B}_{\rho(i)} \alpha$, then $\Gamma, \Sigma \mathbb{B}_{\rho(i)} \alpha$.

Reflexivity guarantees that the base set will be included in bel , and the closure condition establishes closure of bel under derivation.

The chief motivation for requiring derivational closure is that it simplifies the technical task of formalizing the deduction model. Consider the problem of formalizing a belief subsystem that has a complex control strategy guiding its inferential process. To do this correctly, one must write axioms that not only describe the agendas, proof trees, and other data structures used by the control strategy, but also describe how the control strategy guides inference rules operating on these structures. Reasoning about the inference process involves using these axioms to perform deductions that *simulate* the belief inference process, a highly inefficient procedure. By contrast, the assumption of derivational closure leads to a simple formalization of deduction structures in a logic B that incorporates the belief inference process in a direct way. We need not differentiate between a belief as a member of the base set, or as a derived sentence. A sentence that follows from any members of the belief set is itself a belief. The axiomatization of B is simplified, since we need only have an operator whose intended interpretation is membership in the belief set. In Section 4, we exploit the properties of closed derivational systems to exhibit a complete axiomatization of B , using techniques that are manner similar to the *procedural attachment* methods of Weyhrauch [43].

The closure property is an extremely important one, and we should examine its repercussions closely. A point that we have already made is that derivational closure is not the same as consequential closure. The latter refers to a property of sets of sentences based on their *semantics*: every logical consequence of the set is also a member of the set. The former refers to the *syntactic* process of derivability; if the rules \mathcal{R} are not logically complete, then a set of sentences that is derivationally closed under \mathcal{R} need not be consequentially closed.

One of the key properties of belief subsystems that we wish to model is the incompleteness of deriving the consequences of the base set of beliefs. We have identified three sources of incompleteness in belief subsystems: an agent's belief inference rules may be too weak from a logical standpoint, or he may decide that some beliefs are irrelevant to a query, or his control strategy may perform only a subset of the inferences possible when confronted with resource limitations. The assumption of derivational closure for deduction structures affects their ability to model incomplete control strategies, since closure demands that all possible deductions be performed in deriving the belief set.

For an important class of incomplete control strategies, however, there is a corresponding complete control strategy operating on a different set of inference rules that produces the same beliefs on every base set. The criteria that defines this class is that the control strategy use only a *local cost bound* in deciding to drop a particular line of inference. By "local" is meant that the control strategy will always pursue a line of inference to a certain point, without regard to other lines of inference it may be pursuing in parallel. Control strategies with a local cost bound are important because their inferential behavior is predictable: all inferences of a certain sort are guaranteed to be made.

Deduction structures can accurately model the class of locally bounded incomplete control strategies by using an appropriate set of logically incomplete deduction rules. A good example is found in the solution to the chess problem in Section 5. The agent's control strategy applies general rules about chess to search the game tree to only a limited depth; this is modeled in a deduction structure by using deduction rules that work only above a certain depth of the game tree, and applying them exhaustively.

In belief subsystems whose control strategies have a global cost bound, the concept of belief itself is complicated, since one must differentiate between base beliefs and beliefs inferred with some amount of effort. Deduction structures are only an approximate model of these subsystems, and a language with a single belief operator is no longer sufficient for their axiomatization.

Recursion Property

If belief subsystems adhere to the recursion property, then agents view other agents as having belief subsystems similar to their own. This still leaves a considerable degree of

flexibility in representing nested beliefs. For example, an agent John might believe that Sue's internal language is L_1 and that she has a set of deduction rules \mathcal{R}_1 , whereas Kim's internal language is L_2 and her deduction rules are \mathcal{R}_2 . In addition, John might believe that Sue believes that Kim's internal language is L_3 , and that her rules are \mathcal{R}_3 . We call the description of a belief subsystem at some level of nesting a *view*; formally, views are sequences of agents' names, so that the view $John, Sue$ is Sue's belief subsystem as John sees it. We will often use the Greek letter ν to stand for an arbitrary view, and lowercase Latin letters (i, j , etc.) for singleton views, which are agents' actual belief subsystems. Since the formal objects of the deduction model are deduction structures, these will be indexed by views when appropriate. For example, the $d_{John, Sue}$ is a deduction structure modeling the view $John, Sue$.

Obviously, some fairly complicated and confusing situations might be described, with agents believing that other agents have belief subsystems of varying capabilities. Some of these scenarios would be useful in representing situations that are of interest to AI systems; e.g., an expert system tutoring a novice in some domain would need a representation of the novice's deductive capabilities that would initially be less powerful and complete than its own, and could be modified as the novice learned about the domain.

We model the recursion property of belief subsystems within the framework of deduction structures by allowing sentences of L to refer to the beliefs of agents. A standard construct is to have a *belief operator* in L : an operator whose arguments are an agent S and a sentence P , and whose intended meaning is that S believes P . According to the recursion property, this means that the belief operator must have a deduction structure as its interpretation. Deduction rules that apply to belief operators will be judged sound if they respect this interpretation. For example, suppose a deduction structure d_ν has a rule stating that the sentence "John believes q " can be concluded from the premise sentences "John believes p " and "John believes $p \supset q$ ". This is a sound rule of d_ν if *modus ponens* is believed to be a rule of John's belief subsystem as viewed from the view ν , since the presence of p and $p \supset q$ in a deduction structure with *modus ponens* means that q will be derived.

Several simplifying assumptions are implicit in the use of deduction structures to model the nested views of belief subsystems. The language L contains a belief operator

that denotes membership in a belief set (its intended interpretation), and so L can describe what sentences are contained in an agent's belief set. However, there is no provision in L for talking about the deduction rules an agent uses. Instead, these nested-belief rules are implicitly specified by the rules that manipulate sentences with belief operators. Consider the example from the previous paragraph. Let us suppose that we are modeling Sue's belief subsystem with the deduction structure d_{Sue} . Because Sue believes that John uses *modus ponens*, a sound rule of inference for d_{Sue} would be the one that was stated above, namely, the sentence "John believes q " could be concluded from the premise sentences "John believes p " and "John believes $p \supset q$." All of the rules that Sue believes John uses are modeled in this way. Similarly, if, in Sue's opinion, John believes that Kim uses a certain rule, this will be reflected in a rule of John's deduction structure as seen by Sue, which in turn will be modeled by a rule in d_{Sue} . The deduction model thus assumes that the rules for each view, though they may be different, are a fixed parameter of the model. We introduce the function $\rho(\nu)$ to specify deduction rule sets for each view ν ; thus, for each function ρ and each language L , there is a class of deduction structures $D(L, \rho)$ that formalize the deduction model. If the rules ρ are complete with respect to the semantics of L , then the class is said to be *saturated*, and is written $D_s(L, \rho)$.

A final simplification that is not inherent in the deduction model, but which we introduce here solely for technical convenience, is to assume that all deduction structures in all views use the same language L . There are situations in which we might want to relax this restriction, it makes the axiomatization less complex in dealing with the problems at hand.

4. The Logic Family B

We now define a family of logics $B(L, \rho)$ for stating facts and reasoning about deduction structures. This family is parameterized in the same way as deduction structures, namely by an agents' language L and an ensemble of deduction structure rules ρ . Each logic of the family is an axiomatization of the deduction structures $D(L, \rho)$.

The language of B includes operators for stating that sentences are beliefs of an agent, but not for describing deduction rules of agents. Thus the deduction rules are a parameter of the logic family, and are fixed once we decide to use a particular logic of the family. The ensemble function ρ picks out a set of rules for each agent. The reason we chose to make the deduction rules a parameter of B is that it is then possible to find efficient proof methods for B. One of the interesting features of B's axiomatization is that agents' rules are actually present as a subset of the rules of B; proofs about deduction structures in B use these rules directly in their derivation.

The logic of B is framed in terms of a modified form of Gentzen systems, the block tableau systems of Hintikka. Although they may be unfamiliar to some readers, block tableaux are easy to work with and possess some natural advantages when applied to the formalization of deduction structures. Unlike Hilbert systems, which contain complex logical axioms and a single rule of inference in the propositional case (*modus ponens*), block tableau systems have simple axioms and a rich and flexible method of specifying deduction rules. We exploit this capability when we incorporate deduction structure rules into B.

In this section we first present a brief overview of block tableaux. Then we give the postulates of the family B, and a particularly simple subfamily called BK that will be used in solving the problems. By way of example, we prove some theorems of BK.

4.1. Block Tableaux

Most of this section will comprise a review for those readers who are already familiar with tableaux systems.

The Base Language L_0

The language of \mathcal{B} is formed from a base language L_0 that does not contain any operators referring to beliefs. L_0 is taken to be a first-order language with constant terms. An interpretation of L_0 is a truth-value assignment to all sentences (closed formulas) of L_0 ; this assignment must be a *first-order valuation*, that is, it must respect the standard interpretation of the universal and existential quantifiers as well as the Boolean connectives.

We call L_0 *uninterpreted* if every first-order valuation is an interpretation of L_0 ; *partially interpreted* if some proper subset of the first-order valuations are interpretations of L_0 ; and *fully interpreted* (or simply *interpreted*) if there is a singleton interpretation of L_0 . A sentence of L_0 is *valid* if and only if it is true in every interpretation of L_0 .

We use lowercase Latin or Greek letters (p, q, α , etc.) as metavariables that stand for sentences of L_0 . A formula of L_0 that possibly contains the free variable x will be indicated by $\alpha(x)$; the formula derived by substituting the constant a everywhere for x is denoted by $\alpha(x/a)$. Uppercase Greek letters ($\Gamma =_{df} \{\gamma_1, \gamma_2, \dots\}$, $\Delta =_{df} \{\delta_1, \delta_2, \dots\}$, etc.) stand for *finite sets* of sentences of L_0 . By p, Γ we mean the set $\{p\} \cup \Gamma$. We also introduce the abbreviation $\neg\Gamma =_{df} \{\neg\gamma_1, \neg\gamma_2, \dots\}$.

Sequents

Sequents are the main formal object of block tableaux systems.

DEFINITION 4.1. A *sequent* is an ordered pair of finite sets of sentences, (Γ, Δ) . This sequent will also be written as $\Gamma \Rightarrow \Delta$, and read as " Δ follows from Γ ."

A sequent $\Gamma \Rightarrow \Delta$ is true in an interpretation of its component sentences iff one of γ_i is false, or one of δ_j is true. A sequent is *valid* iff it is true under all interpretations, and *satisfiable* iff it is true in at least one interpretation.

From the definition of truth for a sequent, it should be clear that a sequent $\Gamma \Rightarrow \Delta$ is true in an interpretation just in case the sentence $(\gamma_1 \wedge \gamma_2 \wedge \dots) \supset (\delta_1 \vee \delta_2 \vee \dots)$ is true in that interpretation. Thus, in a given interpretation a true sequent can be taken as asserting that the conjunction of γ 's materially implies the disjunction of the δ 's.

We allow the empty set ϕ to appear on either side of a sequent, and abbreviate $\phi \Rightarrow \Delta$ by $\Rightarrow \Delta$, $\Gamma \Rightarrow \phi$ by $\Gamma \Rightarrow$, and $\phi \Rightarrow \phi$ by \Rightarrow . By the above definition, $\Rightarrow \Delta$ is true (in an interpretation) if and only if one of δ_i is true, $\Gamma \Rightarrow$ is true if and only if one of γ_i is false, and \Rightarrow is never true in any interpretation.

Block Tableaux for L_0

The proof method we adopt is similar to Gentzen's original sequent calculus, but simpler in form. It is called the *method of block tableaux*, and was originated by Hintikka [13]. A useful reference is Smullyan [40], in which many results in block tableaux and similar systems are presented in a unified form.

A block tableau system consists of axioms and rules (collectively, *postulates*) whose formal objects are sequents. Block tableau rules are like upside-down inference rules: the conclusion comes first, next a horizontal line, then the premises. Block tableaux themselves are derivations whose root is the sequent derived, whose branches are given by the rules, and whose leaves are axioms. Block tableaux look much like upside-down Gentzen system trees. (A more formal definition of block tableaux is given below).

We consider a system \mathcal{T}_0 (see Smullyan [40], pp. 105–109) that is first-order sound and complete: its consequences are precisely the sentences true in every first-order valuation.

DEFINITION 4.2. *The system \mathcal{T}_0 has the following postulates.*

Axioms. $\Gamma, p \Rightarrow \Delta, p$

Conjunction Rules. $C_1: \frac{\Gamma, p \wedge q \Rightarrow \Delta}{\Gamma, p, q \Rightarrow \Delta}$

$C_2: \frac{\Gamma \Rightarrow \Delta, p \wedge q}{\Gamma \Rightarrow \Delta, p \quad \Gamma \Rightarrow \Delta, q}$

Disjunction Rules.	$D_1 :$	$\frac{\Gamma \Rightarrow \Delta, p \vee q}{\Gamma \Rightarrow \Delta, p, q}$	
	$D_2 :$	$\frac{\Gamma, p \vee q \Rightarrow \Delta}{\Gamma, p \Rightarrow \Delta \quad \Gamma, q \Rightarrow \Delta}$	
Implication Rules.	$I_1 :$	$\frac{\Gamma \Rightarrow \Delta, p \supset q}{\Gamma, p \Rightarrow \Delta, q}$	
	$I_2 :$	$\frac{\Gamma, p \supset q \Rightarrow \Delta}{\Gamma \Rightarrow \Delta, p \quad \Gamma, q \Rightarrow \Delta}$	
Negation Rules.	$N_1 :$	$\frac{\Gamma \Rightarrow \Delta, \neg p}{\Gamma, p \Rightarrow \Delta}$	
	$N_2 :$	$\frac{\Gamma, \neg p \Rightarrow \Delta}{\Gamma \Rightarrow \Delta, p}$	
Universal Rules.	$U_1 :$	$\frac{\Gamma, \forall x. \alpha(x) \Rightarrow \Delta}{\Gamma, \alpha(x/a), \forall x. \alpha(x) \Rightarrow \Delta}$	
	$U_2 :$	$\frac{\Gamma \Rightarrow \forall x. \alpha(x), \Delta}{\Gamma \Rightarrow \alpha(x/a), \forall x. \alpha(x), \Delta},$	where a has not appeared in the tableau
Existential Rules.	$E_1 :$	$\frac{\Gamma \Rightarrow \exists x. \alpha(x), \Delta}{\Gamma \Rightarrow \alpha(x/a), \exists x. \alpha(x), \Delta}$	
	$E_2 :$	$\frac{\Gamma, \exists x. \alpha(x) \Rightarrow \Delta}{\Gamma, \alpha(x/a), \exists x. \alpha(x), \Rightarrow \Delta},$	where a has not appeared in the tableau

Remarks. Note the simple form of the axioms and the symmetric nature of the inference rules (actually, each rule is a rule schema, since Γ , Δ , p , q , and α stand for formulas and sets of formulas of L_0). There is one rule that deletes each logical connective on either side of the sequent. For example, the first conjunction rule deletes a conjunction on the left side of a sequent in favor of the two conjoined sentences; informally, it can be read as “ Δ follows from Γ and $p \wedge q$ if it follows from Γ , p , and q .” It is easily verified that each rule is *sound* with respect to first-order valuations: if the premises are true in an interpretation, then so is the conclusion.

DEFINITION 4.3. A *block tableau* for the sequent $\Gamma \Rightarrow \Delta$ in a system \mathcal{T} is a tree whose nodes are sequents, defined inductively as follows.

1. $\Gamma \Rightarrow \Delta$ is the root of the tree.

2. If sequent s is the parent node of daughters $s_1 \dots s_n$,
then $\frac{s}{s_1 \dots s_n}$ is a rule of \mathcal{T} .

A block tableau is closed if all its leaves are axioms. If there is a closed block tableau for the sequent $\Gamma \Rightarrow \Delta$, then this sequent is a theorem of the system \mathcal{T} and we write $\vdash_{\mathcal{T}} \Gamma \Rightarrow \Delta$.

A system \mathcal{T}' is called a subsystem of \mathcal{T} if every rule of \mathcal{T}' is also a rule of \mathcal{T} . If some subsystem \mathcal{T}' of \mathcal{T} has exactly the same theorems as \mathcal{T} , then the rules of \mathcal{T} not appearing in \mathcal{T}' are said to be eliminable from \mathcal{T} , or admissible to \mathcal{T}' .

Block tableaux are similar to the AND/OR trees commonly encountered in AI theorem-proving systems (see Nilsson [32]). Rules C_2 , D_2 , and I_2 cause AND-splitting, while a choice of rules to apply at a tableau node is an OR-split.

Example. Here is a block tableau for the sequent $\exists x. Bx \wedge Ax, \forall x. Cx \supset \neg Bx \Rightarrow \exists x. Ax \wedge \neg Cx$.

$$\begin{array}{c}
 E_2 \quad \frac{\exists x. Bx \wedge Ax, \forall x. Cx \supset \neg Bx \Rightarrow \exists x. Ax \wedge \neg Cx}{\quad} \\
 U_1 \quad \frac{Be \wedge Ae, \forall x. Cx \supset \neg Bx \Rightarrow \exists x. Ax \wedge \neg Cx}{\quad} \\
 E_1 \quad \frac{Be \wedge Ae, Ce \supset \neg Be \Rightarrow \exists x. Ax \wedge \neg Cx}{\quad} \\
 C_1 \quad \frac{Be \wedge Ae, Ce \supset \neg Be \Rightarrow Ae \wedge \neg Ce}{\quad} \\
 I_2 \quad \frac{Ae, Be, Ce \supset \neg Be \Rightarrow Ae \wedge \neg Ce}{\quad} \\
 \begin{array}{cc}
 N_2 \quad \frac{Ae, Be, \neg Be \Rightarrow Ae \wedge \neg Ce}{Ae, Be \Rightarrow Be, Ae \wedge \neg Ce} & C_2 \quad \frac{Ae, Be \Rightarrow Ce, Ae \wedge \neg Ce}{\quad} \\
 \times & N_1 \quad \frac{Ae, Be \Rightarrow Ce, \neg Ce}{Ae, Be, Ce \Rightarrow Ce} \quad \times
 \end{array}
 \end{array}$$

The sequent to be proved is inserted as the root of the tree. By a series of reductions based on the rules of \mathcal{T}_0 , the atoms of the sequent's sentences are extracted from the scope of quantifiers and Boolean operators. Splitting of the tree occurs at the rules I_2 and C_2 ; otherwise the reduction produces just a single sequent below the line. If a tree is found where the sequents at all the leaves are axioms, then the theorem is proved. Note that the logical inferences are from the leaves to the root of the tree, even though we work backwards in forming the tree. At each junction of the tree, the parent sequent is true in an interpretation if all its daughters are true in that interpretation.

An important connection between theoremhood and logical consequence for sequent systems is the following soundness theorem for tableaux.

THEOREM 4.1. *If $\Gamma \Rightarrow p$ is a theorem of \mathcal{T} (where p is a single sentence of L_0), and all the rules of \mathcal{T} are sound, then p is a logical consequence of Γ .*

Proof. If the rules of \mathcal{T} are sound, then every theorem of \mathcal{T} is valid. By Definition 4.1, this means that in every interpretation in which all of Γ are true, p must be also. ■

4.2. The Language of \mathcal{B}

The language of \mathcal{B} is formed from a first-order base language L_0 by adding modal operators for belief and belief circumscription. We call this language $L^{\mathcal{B}}$. It is convenient to use $L^{\mathcal{B}}$ also as the agents' language L , since it provides a representation for nested beliefs as required by the recursion property. With this assumption, we can parameterize \mathcal{B} by the base language L_0 , and write $\mathcal{B}(L_0, \rho)$ for the logic family.

To form $L^{\mathcal{B}}$ from a base language L_0 , we require a countable set of agents (S_0, S_1, \dots).

DEFINITION 4.4. *A sentence of $L^{\mathcal{B}}$ based on L_0 is defined inductively by the following rules.*

1. *All formation rules of L_0 are also formation rules of $L^{\mathcal{B}}$.*
2. *If p is a sentence, then $[S_i]p$ is a sentence for $i \geq 0$.*
3. *If p is a sentence and Γ is a finite set of sentences, then $\langle S_i : \Gamma \rangle p$ is a sentence for $i \geq 1$.*

An ordinary atom of $L^{\mathcal{B}}$ is a ground atom of L_0 ; a belief atom is a sentence of the form $[S_i]p$, and a circumscriptive atom is one of the form $\langle S_i : \Gamma \rangle p$. In the belief atom $[S_i]p$, p is said to be in the context of the belief operator. Note that there is no quantification into the contexts of belief atoms, since the argument of a belief operator is always a closed sentence. $L^{\mathcal{B}}$ can be extended to include quantification into belief contexts: such a language has greater representational power and its logic $q\mathcal{B}$ has a more complex axiomatization. The interested reader is referred to Konolige [21] for a description of $q\mathcal{B}$. Here, the simpler \mathcal{B} is sufficient for an analysis of the problems.

We will use the abbreviation $[S]\Gamma =_{df} [S]\gamma_1, [S]\gamma_2, \dots$

Interpretations

Interpretations of the language of L^B are formed from interpretations of its base language L_0 , together with an interpretation of belief and circumscriptive atoms. The intended meaning of the belief atom $[S_i]p$ is that p is in the belief subsystem of agent S_i ; informally, we would say " S_i believes p ." Since we are formalizing belief subsystems by means of deduction structures, an interpretation of the belief atoms $[S_i]p$ is given by a deduction structure d_i . $[S_i]p$ is true if p is in $\text{bel}(d_i)$, the belief set of d_i ; otherwise it is false.

In addition to representing beliefs of individuals, we use belief atoms to represent common beliefs. A common belief is one that every agent believes, and every agent believes every other agent believes, and so on to arbitrary depths of belief nesting. We reserve the name S_0 for a fictional agent whose beliefs are taken to be common among all agents. The belief atom $[S_0]p$ means that p is a common belief. In terms of deduction structures, its intended interpretation is that p and $[S_0]p$ are in the deduction structure d_i of every agent S_i , $i \geq 0$.

McCarthy (see, for example, [25]) was the first to recognize the common knowledge could be represented by the use of a fictitious agent FOOL whose knowledge "any fool" would know. He used a possible-worlds semantics for knowledge, and so all consequences of common knowledge were also known. The representation of common belief presented here uses an obviously similar approach; it differs only in that common belief rather than common knowledge is axiomatized (common beliefs need not be true), and in having a deduction structure semantics, so that common beliefs need not be closed under logical consequence.

The interpretation of circumscriptive atoms is also given by the deduction structure representing an agent's beliefs. The intended meaning of $\langle S_i : \Gamma \rangle p$ is that p is derivable from Γ in the deduction structure d_i , that is, $\Gamma \vdash_{\rho(i)} p$. The circumscription operator elevates the belief derivation process to a first-class entity of the language (as opposed to belief operators $[S_i]$, which simply state that certain sentences are in or not in the belief set).

While it may not be apparent at first glance, the circumscription operator is a powerful tool for representing situations of delimited knowledge. For example, to formally state the condition, "the only facts that agent S knows about proposition p are F ," we could use

$$(4.1) \quad \langle S : F \rangle p \equiv [S]p .$$

This assertion states that S believing p is equivalent to S being able to derive p from F . The forward implication is uninteresting, since it just says that p is derivable from F by agent S , i.e., $[S]F \supset [S]p$. The reverse implication is more interesting, since it states p cannot be a belief of S *unless* it is derivable from F . This reverse implication limits the information S has available to derive p to the sentences F , and thus gives the circumscriptive content of (4.1). Note that there is no way to formulate the reverse implication as a sentence of L^B using only belief operators.

The reader should note carefully that the semantics of L^B differs completely from that of most modal languages, in which the argument to the modal operator is usually taken to denote a *proposition* that can take on a truth-value in a possible world. By contrast, arguments to modal operators in the language of B denote *sentences* of L , namely themselves. It is important to keep this distinction in mind when interpreting the modal operators of B .

4.3. A Sequent System for B

The deductive process that underlies the deduction model is characterized in very general terms by deduction structures and their associated belief sets. Until now we have been content with deliberate vagueness about the exact nature of deduction rules and the derivation process. As stated in Section 3, there are five conditions that must be satisfied: the deduction rules must be *effective*, *provincial*, and *sound*, and the derivations *reflexive* and *closed under deduction*. Consider a deduction structure $d_i = \langle B, \rho(i) \rangle$ for agent S_i . If we let the process of belief derivation for d be symbolized by $\vdash_{\rho(i)}$, these conditions are as follows.

(Effectiveness) The deduction rules $\rho(i)$ are effectively applicable.

- (Provinciality) The number of input sentences to each rule is finite and bounded.
- (Soundness) If $\Gamma \vdash_{\rho(i)} \alpha$, then α is a logical consequence of Γ .
- (Reflexivity) $\alpha \vdash_{\rho(i)} \alpha$.
- (Closure) If $\Gamma \vdash_{\rho(i)} \beta$ and $\beta, \Sigma \vdash_{\rho(i)} \alpha$, then $\Gamma, \Sigma \vdash_{\rho(i)} \alpha$.

Suppose we are given beforehand a derivation operator $\vdash_{\rho(i)}$, satisfying the above conditions, that models an agent S_i 's belief subsystem. The central problem in the formulation of \mathcal{B} is to find tableau rules that correctly implement the meaning of the belief operator $[S_i]$ and the circumscription operator $\langle S_i : \Gamma \rangle$ under $\vdash_{\rho(i)}$.

Consider first the sequent $[S_i]\Gamma \Rightarrow [S_i]\alpha$. Its intended meaning is that, if all of Γ are in S_i 's belief set, then so is α . The only possible way that we can guarantee this condition is if α is derivable from Γ , i.e., $\Gamma \vdash_{\rho(i)} \alpha$. If this were not the case, then we could always construct the counterexample $d_i =_{df} \langle \Gamma, \rho(i) \rangle$ in which all of Γ are in d_i , but α is not. Thus we can relate the truth of a sequent involving belief operators to derivability in an agent's belief subsystem. This relation is captured by the inference rule

$$A : \frac{\Sigma, [S_i]\Gamma \Rightarrow [S_i]\alpha, \Delta}{\Gamma \vdash_{\rho(i)} \alpha}$$

A is called the *attachment rule*, because it derives results involving the belief operator by attaching sentences about belief to the actual derivation process of an agent. Remembering that the premise is the bottom sequent and the conclusion the top, we can read A informally as follows: "If α is a deductive consequence of Γ in S_i 's belief subsystem, then, whenever S_i believes Γ , he also believes α ."

To capture the notion of common belief, we need to make a modification to the attachment rule. The intended meaning of the common belief atom $[S_0]q$ is that both q and $[S_0]q$ are in the belief subsystem of every agent. The sequent $[S_0]\Lambda, [S_i]\Gamma \Rightarrow [S_i]\alpha$ will be true if whenever $[S_0]\Lambda$, Λ , and Γ are in the belief set of d_i , α also is. By reasoning similar to that used in deriving the rule A , we can rephrase this in terms of belief derivation. This yields the revised attachment rule A^{CB} .

$$A^{CB} : \frac{\Sigma, [S_0]\Lambda, [S_i]\Gamma \Rightarrow [S_i]\alpha, \Delta}{[S_0]\Lambda, \Lambda, \Gamma \vdash_{\rho(i)} \alpha}$$

In A^{CB} , both Λ and $[S_0]\Lambda$ can be used in the derivation of α . Note that this rule is applicable to the fictional agent S_0 . Because S_0 's beliefs are intended to be common beliefs, and hence derivable by any agent, it should be the case that the rules $\rho(0)$ are used by every agent. We thus demand that $\rho(0) \subseteq \rho(i)$ for every i .

We can find tableau rules for the circumscription operator in a similar manner. The intended semantics of this operator relates directly to the belief derivation process: $\langle S_i : \Gamma \rangle p$ means that p is derivable from Γ in S_i 's belief subsystem, i.e., $\Gamma \vdash_{\rho(i)} p$. In writing sequent rules, there are two cases to consider, for a circumscriptive atom can appear on the right or left side of the sequent arrow. We thus have the following two rules.

$$Circ_1 : \frac{\Sigma \Rightarrow \langle S_i : \Gamma \rangle p, \Delta}{\Gamma \vdash_{\rho(i)} p}$$

$$Circ_2 : \frac{\Sigma, \langle S_i : \Gamma \rangle p \Rightarrow \Delta}{\Gamma \vdash_{\rho(i)} p}$$

The second circumscription rule is the one that is used to show circumscriptive ignorance. It states that if p is not derivable from a set of sentences Γ , then the circumscriptive atom $\langle S_i : \Gamma \rangle p$ is false. Given a statement of the form 4.1, this in turn would imply that S_i was ignorant of p .

We can now give a full axiomatization of the logic family \mathcal{B} .

DEFINITION 4.5. *The system $\mathcal{B}(L_0, \rho)$ has the following postulates.*

1. *The first-order complete rules \mathcal{T}_0 .*
2. *The rules A^{CB} , $Circ_1$, and $Circ_2$.*
3. *A closed derivation process $\vdash_{\rho(i)}$ for each agent S_i , such that $\rho(0) \subseteq \rho(i)$ for every i .*

This axiomatization of \mathcal{B} is both sound and complete with respect to its deduction structure semantics, as proven in Konolige [21]. It is a compact formalization of the deduction model and useful for theoretical investigations, but we do not use it very much as a representational formalism because of the general nature of the belief deduction process $\vdash_{\rho(i)}$, which is rather opaque to further analysis. For instance, we might wish to look at

the subfamily of \mathcal{B} in which the rules of $\rho(i)$ that govern nested belief are as strong as \mathcal{A} . In order to explore the fine structure of S_i 's belief deduction process, or to formalize the problems, we need to fix the nature of $\mathfrak{B}_{\rho(i)}$ more precisely. The rich set of rules, and the flexibility of tableau derivations, make tableau systems a natural choice here. In the next section we define a particularization of \mathcal{B} , the logic family BK, whose belief derivation process is defined in block tableaux terms.

4.4. The Nonintrospective Logic Family BK

In the logic family BK, the belief derivation operator \mathfrak{B} is defined as provability in a tableau system.

DEFINITION 4.6. *A sentence α is BK-derivable from premises Γ ($\Gamma \mathfrak{B}_T \alpha$) if and only if $\vdash_T \Gamma \Rightarrow \alpha$.*

We need to show that tableau system derivability as just defined satisfies the five criteria of belief derivation: effectiveness, provinciality, soundness, reflexivity and closure. Consider a sequent system \mathcal{T} made up of sound tableau rules. According to Theorem 4.1, the theorem $\vdash_T \Gamma \Rightarrow p$ of \mathcal{T} implies that p is a logical consequence of Γ , so we are assured that \vdash_T satisfies the soundness criterion. Provinciality and effectiveness are also satisfied, since the theorems of \mathcal{T} are built by using effectively computable steps that operate on a bounded number of sentences at each step. The observant reader might object at this point that tableau rules may indeed refer to an unbounded number of premise sentences; e.g., any of the rules of \mathcal{T}_0 have this property, since Γ and Δ can stand for any set of sentences. However, each rule of \mathcal{T}_0 is actually a rule schema: the capital Greek letters are metavariables that are instantiated with a boundedly finite set of sentences to define a rule.

The closure condition is fulfilled by a special subclass of sequent systems, namely those for which the following rule, Cut^* , is admissible:

$$Cut^* : \frac{\Gamma, \Sigma \Rightarrow \alpha}{\Gamma \Rightarrow \beta \quad \beta, \Sigma \Rightarrow \alpha} .$$

To see how this rule guarantees closure, suppose that $\Gamma \Rightarrow \beta$ and $\beta, \Sigma \Rightarrow \alpha$ are both theorems of a sequent system \mathcal{T} for which Cut^* is admissible. Because Cut^* is admissible

and both of its premises have closed tableaux, the conclusion $\Gamma, \Sigma \Rightarrow \alpha$ must also be a theorem.

Finally, the derivation process will be reflexive ($\alpha \vdash_T \alpha$) if we include the following axiom in the system T :

$$Id: \quad \Sigma, \alpha \Rightarrow \alpha, \Delta$$

Thus we only allow a system T to appear in a deduction structure $d(B, T)$ if the system is sound, Cut^* is an admissible rule of T , and Id is an axiom of T .

An interesting consequence of using tableau derivations in BK is that the attachment rule A can now be expressed wholly in terms of sequents, eliminating the derivation operator. To see how this comes about, consider first replacing the belief operator in rule A by tableau provability, as given by Definition 4.6. This yields

$$AK': \quad \frac{\Sigma, [S_i]\Gamma \Rightarrow [S_i]\alpha, \Delta}{\vdash_{r(i)} \Gamma \Rightarrow \alpha},$$

where $r(i)$ is the set of tableau rules used by agent S_i .

Now $\vdash_{r(i)} \Gamma \Rightarrow \alpha$ is true precisely if there is a closed tableau for $\Gamma \Rightarrow \alpha$, using the rules $r(i)$. Hence we should be able to eliminate the provability symbol if we add the rules $r(i)$ to B for the purpose of constructing a tableau for $\Gamma \Rightarrow \alpha$. In order to keep the agents' rules $r(i)$ from being confused with the rules of B , we add an agent index to sequents to indicate that the tableau rules to be use are for a particular agent. The final version of the attachment rule is

$$AK: \quad \frac{\Sigma, [S_i]\Gamma \Rightarrow [S_i]\alpha, \Delta}{\Gamma \Rightarrow_i \alpha}$$

Agents' rules are expressed using the indexed sequent sign. e.g., if agent S_i were to use C_2 , the following rule would be added to B .

$$C_2^i: \quad \frac{\Gamma \Rightarrow_i \Delta, p \wedge q}{\Gamma \Rightarrow_i \Delta, p \quad \Gamma \Rightarrow_i \Delta, q}$$

Taking the recursion property of belief subsystems seriously, we can iterate the process just described for the attachment rule. Each agent treats other agents as having

a set of tableau rules. In formulating BK, there will be a tableau rule set associated with each view (views are discussed in relation to the recursion property in Section 3.3). Let us symbolize the set of tableau rules representing the view ν by $\tau(\nu)$.

A sequent $\Gamma \Rightarrow_{\nu} \Delta$, with index ν , is a statement about the belief subsystem of the view ν . For example, if $\nu = \text{Sue, Kim}$, the sequent $\Gamma \Rightarrow_{\nu} p$ states that p follows from Γ in Sue's view of Kim's belief subsystem. The deduction rules $\tau(\nu)$ always have sequents indexed by ν in their conclusions (above the line). This assures us that they will always be used as rules of the belief subsystem ν , and of no other.

The logic BK can thus be parameterized by a set of tableau rules for each view, and we write $\text{BK}(L_0, \tau)$ to indicate this. If the sequent $\Gamma \Rightarrow_{\nu} \Delta$ is a theorem of the logic $\text{BK}(L_0, \tau)$, it asserts that the sequent $\Gamma \Rightarrow \Delta$ is provable in the view ν . We write this as $\vdash_{\text{BK}(L_0, \tau)} \Gamma \Rightarrow_{\nu} \Delta$. If this sequent is a theorem for every parameterization of BK, we write simply $\vdash \Gamma \Rightarrow_{\nu} \Delta$. Note that the presence of the index on the sequent means that we do not have to state explicitly that the set of rules used to derive the theorem were those of the view ν . Properties of the the actual belief subsystems are always stated using unindexed sequent; for example, to show formally that if an agent believes p , then he believes q , we would have to prove that the sequent $[S_i]p \Rightarrow [S_i]q$ is a theorem of BK.

Postulates of $\text{BK}(L_0, \tau)$

This family is parameterized by a base language L_0 and tableau rules $\tau(\nu)$ for each view ν .

DEFINITION 4.7. *The system $\text{BK}(L_0, \tau)$ is given by the following postulates:*

1. *The first-order complete rules τ_0 .*
2. *The attachment rule*

$$AK^{CB} : \frac{\Sigma, [S_0]\Lambda, [S_i]\Gamma \Rightarrow [S_i]\alpha, \Delta}{[S_0]\Lambda, \Lambda, \Gamma \Rightarrow_i \alpha}$$

3. *A set of sound sequent rules $\tau(\nu)$ for each view ν which contains the axiom Id , and for which the rule Cut^* is admissible. Also, $\tau(\nu, 0) \subseteq \tau(\nu, i)$ for all views ν and agents S_i .*

4. The circumscription rules

$$CircK_1 : \frac{\Sigma \Rightarrow \langle S_i : \Gamma \rangle p, \Delta}{\Gamma \Rightarrow_i p}$$

and

$$CircK_2 : \frac{\Sigma, \langle S_i : \Gamma \rangle p \Rightarrow \Delta}{\not\vdash \Gamma \Rightarrow_i p}$$

Remarks. There are three parts to the system $BK(L_0, \tau)$. The first part is a set of rules that perform first-order deductions about the real world. These rules incorporate the unsubscripted sequent sign (\Rightarrow).

The second part is the attachment rule AK^{CB} , together with a set of rules formalizing the deductive system of each view. These rules involve the sequent sign \Rightarrow_ν , since they talk about agents' deductive systems. They can contain rules that have a purely nonmodal import (e.g., rules of \mathcal{T}_0), as well as rules that deal with belief operators. The rule Cut^* , which implements the closure property of belief sets, must be an admissible rule of $\tau(\nu)$.

The rules $\tau(\nu)$ of a view ν can be incomplete in several ways. They may be first-order incomplete, in which case they cannot be used to draw all the consequences of sentences involving nonmodal operators that they otherwise might (to be first-order complete, it is sufficient for the rules \mathcal{T}_0 to be admissible in a view). They may also be incomplete with respect to the semantics of sentences involving belief operators. To be complete in this respect, a sufficient rule would be AK^{CB} . A view for which this rule is admissible is called *recursively complete*. If every view of a logic $BK(L_0, \tau)$ is recursively and first-order complete, the logic is called *saturated*. We will symbolize the subfamily of saturated logics by BK_s .

The rule AK^{CB} is a weak version of the attachment rule A^{CB} in that it makes no assumptions about the beliefs an agent may have of his own beliefs. For example, we might argue that, if an agent S believes a proposition P , then he believes that he believes it. All he has to do to establish this is query his belief subsystem with the question, "Do I believe P ?" If the answer comes back "yes," then he should be able to infer that he does indeed believe P , i.e., $[S][S]P$ is true if $[S]P$ is. However, as far as rule AK^{CB} is concerned, an

agent's own belief subsystem has the same status for him as does that of any other agent. In particular, AK^{CB} allows an agent to have false and incomplete beliefs about his own beliefs. Other version of AK^{CB} with stronger assumptions about self-belief are possible (see Section 6).

The third part consists of the two circumscription rules. The provability operator can be eliminated from $CircK_1$, but not from $CircK_2$. In order to show that p does not follow from Γ for S_i , we must show that there is no closed tableau for $\Gamma \Rightarrow_i p$. One technique that we use in solving the problems is the following. If there is no closed tableau for a saturated logic of BK, there is no closed tableau for any logic of BK. Every theorem of saturated BK is a theorem of the normal modal system $K4$ (see Section 6), which has a decision procedure based on the methods of Sato (in [38]). Thus if a sequent is not provable in $K4$, it is not provable in any logic of BK.

Some Theorems of BK

THEOREM 4.2. *Let p be derivable from Γ in the view i of $BK(L_0, \tau)$. Then*

$$\vdash_{BK(L_0, \tau)} [S_i]\Gamma \Rightarrow [S_i]p .$$

Proof. In one step, using rule AK^{CB} :

$$AK^{CB} \frac{[S_i]\Gamma \Rightarrow [S_i]p}{\Gamma \Rightarrow_i p} \quad \times$$

■

THEOREM 4.3. *Let ν be a recursively complete view of $BK(L_0, \tau)$, and let p be derivable from Γ in the view ν, i . Then*

$$\vdash_{BK(L_0, \tau)} [S_i]\Gamma \Rightarrow_\nu [S_i]p .$$

Proof. In one step, using rule AK^{CB} of $\tau(\nu)$:

$$AK^{CB} \frac{[S_i]\Gamma \Rightarrow_\nu [S_i]p}{\Gamma \Rightarrow_{\nu, i} p} \quad \times$$

Remarks. These two theorems show that BK has a weakened analog of the necessitation rule of modal logic (if α is provable, so is $\Box\alpha$). If a nonmodal sentence α is provable in the view i (i.e., $\vdash_{BK(L_0, r)} \alpha$), then, by Theorem 4.2, $[S_i]\alpha$ is provable in the empty view. Since the theorems of $r(i)$ are assumed to be sound, α is a tautology, and so must be provable in the empty view.¹ Hence, for those tautologies provable in the view i , necessitation holds. Theorem 4.3 establishes this result for an arbitrary view in which A is an admissible rule. Depending on the exact nature of the rule sets r , necessitation will hold for some subset of the provable sentences of a particular logic $BK(L_0, r)$.

THEOREM 4.4. $\nvdash [S_i]p \Rightarrow p$

Proof. If p is a primitive sentence, then there is no applicable tableaux rule, and hence no closed tableaux for the sequent. ■

Remarks. The familiar modal logic principle $\Box p \supset p$ (if p is necessary, then p is true) is not a theorem of BK, since beliefs need not be true.

THEOREM 4.5. $\nvdash [S_i]p \Rightarrow [S_i][S_i]p$

Proof. The only applicable rule is AK^{CB} :

$$AK^{CB} \frac{[S_i]p \Rightarrow [S_i][S_i]p}{p \Rightarrow_i [S_i]p}$$

According to the semantics of the deduction model, the sequent $p \Rightarrow_i [S_i]p$ is not valid: just because a sentence p is true does not mean that an agent S_i believes it. Hence, there cannot be any set of sound tableau rules for $r(i)$ that causes $p \Rightarrow_i [S_i]p$ to close. ■

¹ Care must be taken in restricting α to nonmodal sentences, since the semantics of modal operators can change from one view to another (see the discussion of the recursion property in Section 3.3). John may believe perfectly well that Sue's belief subsystem can prove a certain fact, whereas in actuality her inference rules are too weak.

THEOREM 4.6. $\nvdash \neg[S_i]p \Rightarrow [S_i]\neg[S_i]p$

Proof. We can apply either N_2 or AK^{CB} . If we apply the latter, we obtain

$$AK^{CB} \frac{\neg[S_i]p \Rightarrow [S_i]\neg[S_i]p}{\Rightarrow_i \neg[S_i]p}$$

deduction model, since it would require that no agent believe any sentence. Hence there can be no set of sound tableau rules $\tau(i)$ that derives it.

If we apply N_2 first, we obtain

$$N_2 \frac{\neg[S_i]p \Rightarrow [S_i]\neg[S_i]p}{\Rightarrow [S_i]p, [S_i]\neg[S_i]p}$$

There are now two ways to apply AK^{CB} . In one application, we generate the sequent $\Rightarrow_i \neg[S_i]p$, which cannot close. In the other, we generate $\Rightarrow_i p$, which again cannot be derived by any set of sound tableau rules. ■

Remarks. These theorems show that no logic of BK sanctions inferences about self-beliefs. If an agent believes p , it does not follow that his model of his own beliefs includes p ; this is the import of Theorem 4.5. Similarly, if he does not believe p , he also may not have knowledge of this fact, as shown by Theorem 4.6.

THEOREM 4.7.

$$\vdash [S_0]p \Rightarrow [S_0][S_0]p$$

Proof.

$$AK^{CB} \frac{[S_0]p \Rightarrow [S_0][S_0]p}{[S_0]p, p \Rightarrow_i [S_0]p}$$

x

■

Remarks. We have proven a simple fact about common beliefs: if p is a common belief, it is a common belief that this is so.

For the circumscriptive ignorance part of BK, it is an interesting exercise to show that

$$(4.2) \quad \langle S_i : \Gamma \rangle p \Rightarrow [S_i] \Gamma \supset [S_i] p$$

holds, but the converse doesn't. That is, if p follows from Γ for agent S_i , it must be the case that believing Γ entails believing p ; on the other hand, it may be that every time an agent has Γ in his base set he also has p , which would satisfy $[S_i] \Gamma \supset [S_i] p$ without having p derivable from Γ .

THEOREM 4.8. $\vdash \langle S_i : \Gamma \rangle p \Rightarrow [S_i] \Gamma \supset [S_i] p$

Proof. We have the following two tableaux for this sentence.

$$\begin{array}{c} I_1 \\ AKCB \end{array} \frac{\langle S_i : \Gamma \rangle p \Rightarrow [S_i] \Gamma \supset [S_i] p}{\frac{\langle S_i : \Gamma \rangle p [S_i] \Gamma \Rightarrow [S_i] p}{\Gamma \Rightarrow_i p}}$$

$$\begin{array}{c} I_1 \\ Circ_2 \end{array} \frac{\langle S_i : \Gamma \rangle p \Rightarrow [S_i] \Gamma \supset [S_i] p}{\frac{\langle S_i : \Gamma \rangle p [S_i] \Gamma \Rightarrow [S_i] p}{\nvdash \Gamma \Rightarrow_i p}}$$

Either p is derivable from Γ using the rules $r(i)$, or it isn't. In either case one of these tableaux closes. ■

Example. we give an example of the use of the circumscription rules to show ignorance. Suppose the agent Sue believes only the sentences P and $P \supset Q$ in a situation; we want to show that she doesn't believe R . Thus we want to prove the sequent $\langle Sue : P, P \supset Q \rangle R \equiv [Sue] R \Rightarrow \neg[Sue] R$.

$$\begin{array}{c} C_1 \\ I_2 \\ Circ_2 \end{array} \frac{\frac{\langle Sue : P, P \supset Q \rangle R \equiv [Sue] R \Rightarrow \neg[Sue] R}{\langle Sue : P, P \supset Q \rangle R \supset [Sue] R, [Sue] R \supset \langle Sue : P, P \supset Q \rangle R \Rightarrow \neg[Sue] R}}{\frac{\langle Sue : P, P \supset Q \rangle R \Rightarrow \neg[Sue] R}{\nvdash P, P \supset Q \Rightarrow_{Sue} R}} \quad \begin{array}{c} N_2 \\ \Rightarrow [Sue] R, \neg[Sue] R \\ [Sue] R \Rightarrow [Sue] R \end{array}$$

x

If the rules $r(Sue)$ are sound, there is no closed tableau for $P, P \supset Q \Rightarrow_i R$, and so both branches of the tableau close. Note that only the reverse implication half of the equivalence was needed.

5. The Problems Revisited

Using the logic BK, we present formal solutions to the two representational problems posed at the beginning of this section. In each case we have tried to avoid solutions that are trivial in the sense that they solve the representational problem, but only at the expense of excluding types of reasoning that might be expected to occur. For example, in the chess problem it would be an adequate but unrealistic solution to credit each player with no deduction rules at all. Instead, we try to find rules that allow a resource-limited amount of reasoning about the game to take place.

The Chess Problem

To approach this problem, we need to represent the game in a first-order language. Because the ontology of chess involves rather complicated objects (pieces, board positions, moves, histories of moves) we will not give a complete formalization, but rather sketch in outline how this might be done.

We use a multisorted first-order language L_c for the base language L_0 . The key sorts will be those for players (S_w or S_b), moves, and boards. The particular structure of the sort terms is not important for the solution of this problem, but they should have the following information. A board contains the position of all pieces, and a history of the moves that were made to get to that position. This is important because we want to be able to find all legal moves from a given position; to do this, we have to have the sequence of moves leading up to the position, since legal moves can be defined only in terms of this sequence. For example, castling can only occur once, even if a player returns to the position before the castle; more importantly, there are no legal moves if 50 moves have been made without a capture or pawn advancement (this is what makes chess a finite game). A move contains

enough information so that it is possible to compute all successor boards, that is, those resulting from legal moves.

The game tree is a useful concept in exploring game-playing strategies. This is a finite tree (for finite games like chess) whose nodes are board positions, and whose branches are all possible complete games. A terminal node of the tree ends in either a win for White or Black, or a draw. The *game-theoretic value* of a node for a player is either 1 (a win), 0 (a draw), or -1 (a loss), based on whether that player can force a win or a draw, or his opponent can force a win. We use the predicate $M(p, b, k, l, r)$ to mean that board b has value k for player p . The argument l is a depth-of-search indicator, and shows the maximum depth of the game tree that the value is based on. We include the argument r so that M can represent heuristic information about the value of a node; when $r = f$, k is the player's subjective estimate of the value of the node, i.e., he has not searched to all terminal nodes of the game tree. If $r = t$, then k is the game-theoretic value of the board.

We take the formal interpretation of boards, players, and the M predicate to be the game of chess, so that L_c is a partially interpreted language. The rules of the game of chess strictly specify what the game tree and its associated values will be; hence, each predication $M(p, b, k, l, t)$ or its negation is a valid consequence of these interpretations. Any agent who knows the rules of chess, and who has the concept of game trees, will know the game-theoretic value of every node if his beliefs are consequentially closed. In particular, he will believe either $M(S_w, I, 1, k, t)$ or $\neg M(S_w, I, 1, k, t)$, where I is the initial board; and so he will know whether White has an initial forced win or not.

We represent agents' knowledge of chess by giving tableau rules for L_c . The rules \mathcal{T}_c presented below are one possible choice.

$$Ch_1 : \frac{\Gamma \Rightarrow M(p, b, k, l, r), \Delta}{\Gamma \Rightarrow M(p, b_1, k_1, l_1, r_1), \Delta \quad \Gamma \Rightarrow M(p, b_2, k_2, l_2, r_2), \Delta \quad \dots \quad \Gamma \Rightarrow M(p, b_n, k_n, l_n, r_n), \Delta}$$

where b_1 - b_n are all the legal successor boards to b
 p 's opponent is to move on b
 k is the minimum of k_1 - k_n
 l is $1 +$ the maximum of l_1 - l_n
 r is t iff all of r_1 - r_n are t

$$Ch_2 : \frac{\Gamma \Rightarrow M(p, b, k, l, r), \Delta}{\Gamma \Rightarrow M(p, b_1, k_1, l_1, r_1), \Delta \quad \Gamma \Rightarrow M(p, b_2, k_2, l_2, r_2), \Delta \quad \cdots \quad \Gamma \Rightarrow M(p, b_n, k_n, l_n, r_n), \Delta}$$

where b_1-b_n are all the legal successor boards to b
 p is to move on b
 k is the maximum of k_1-k_n
 l is 1+ the maximum of l_1-l_n
 r is t iff all of r_1-r_n are t

$$Ch_3 : \Gamma \Rightarrow M(p, b, k, 0, t), \Delta, \quad \text{where } k = 1 \text{ if } p \text{ has a checkmate on his opponent on board } b; k = 0 \text{ if board } b \text{ is a draw; and } k = -1 \text{ if } p\text{'s opponent has a checkmate.}$$

$$Ch_4 : \Gamma \Rightarrow M(p, b, k, 0, f), \Delta, \quad \text{where } k \text{ is any number between } -1 \text{ and } 1$$

Ch_1 axiomatizes nodes in the game tree where p 's opponent moves. The value of such a node is the minimum of the values of its successor nodes. The argument l is the maximum depth of the subtree searched. r will be t only if all the subtrees have been searched to leaf nodes. Ch_2 is similar to Ch_1 , except p moves, and the maximum of the successor values is chosen.

Ch_3 is the rule for terminal nodes of the tree. Ch_4 is a rule for heuristic evaluation of any node; note that the last argument to M is f , which indicates that a terminal node has not been reached. Each agent may have his own particular heuristics for evaluating nonterminal nodes; we can accommodate this by changing the values for k in Ch_4 .

As an example of the use of these rules, consider the following tableau proof.

(5.1)

$$Ch_1 \frac{\Rightarrow M(S_w, b_1, 1, 0, t) \quad \Rightarrow M(S_w, b, 1, 2, t)}{\Rightarrow M(S_w, b, 1, 2, t)} \quad Ch_2 \frac{\Rightarrow M(S_w, b_2, 1, 1, t) \quad \Rightarrow M(S_w, b_5, 1, 0, t)}{\Rightarrow M(S_w, b_2, 1, 1, t)} \quad \Rightarrow M(S_w, b_5, 1, 0, t)$$

\times \times \times

This is a proof that the board b has a value 1 for White, searching to all terminal nodes. Boards b_1 , b_2 , and b_3 all have value 1, so an application of rule Ch_1 yields that value 1 for b (it is Black's turn to move on b). Boards b_1 and b_5 are terminal nodes that are checkmates for White. There are two legal moves from board b_2 ; one ends in a draw (b_3), the other in a win (b_4) for White. Since it is White's turn to move, rule Ch_2 applies.

The structure of this tableau proof mimics exactly the structure of the game tree from the board b . Indeed, for any subtree of the complete game tree of chess whose root is the board b with value k for player p , there is a corresponding proof of $M(p, b, k, l, t)$ using the rules \mathcal{T}_c . In particular, if one of $M(S_w, I, 1, l, t)$, $M(S_w, I, 0, l, t)$, or $M(S_w, I, -1, l, t)$ is true, there is a proof of this fact. Hence the rules \mathcal{T}_c are sufficient for a player to reason whether White has a forced initial win or not, given an infinite resource bound for derivations. If we model agents as having the rules \mathcal{T}_c , so that $\mathcal{T}_c \subseteq r(\nu)$ for every view ν , the conversation presented at the beginning of this paper would make sense: each agent would believe that everyone knew whether White had a forced initial win.

A simple modification of the rules Ch_1 and Ch_2 can restrict exploration of the entire game tree, while still allowing agents to reason about game tree values using the heuristic axioms Ch_4 , or the terminal node axioms Ch_3 if the game subtree is small. All that is necessary is to add the condition that no rule is applicable when the depth l is greater than some constant N . S_w would still be able to reason about the game to depths less than or equal to N , but he could go no further. In this way, a deductively closed system can represent a resource-limited derivation process. The revised rules are

$$\begin{array}{ll} Ch'_1 & Ch_1, \text{ with the condition that } l \leq N. \\ Ch'_2 & Ch_2, \text{ with the condition that } l \leq N. \end{array}$$

With these rules, the proof of (5.1) would still go through for $N \geq 2$, but a proof of $M(S_w, I, k, l, t)$ could not be found if N were low enough to stop search at a reasonable level of the game tree.

The solution to the chess problem illustrates the ability of the deduction model to represent resource bounds by the imposition of constraints on deduction rules. There are other workable constraints for this problem besides depth cutoff: for example, the number of nodes in the tree being searched could be kept below some minimum. Because the structure of proofs mimics the game tree, any cutoff condition that is based on the game tree could be represented by appropriate deduction rules.

The Not-So-Wise-Man Problem

For this problem we use a base language L_w containing only the three primitive propositions P_1 , P_2 , and P_3 . P_i expresses the proposition that wise man S_i has a white spot on his forehead.

In the initial situation, no one has spoken except the king, who has declared that at least one spot is white. Axioms for this situation are

- (W1) $P_1 \wedge P_2 \wedge P_3$
- (W2) $[S_0](P_1 \vee P_2 \vee P_3)$
- (W3) $(P_i \supset [S_j]P_i) \wedge [S_0](P_i \supset [S_j]P_i), \quad i \neq j, \quad j \neq 0$
- (W4) $(\neg P_i \supset [S_j]\neg P_i) \wedge [S_0](\neg P_i \supset [S_j]\neg P_i), \quad i \neq j, j \neq 0$
- (C1) $\langle S_i : W2-4, P_j, P_k \rangle P_i \equiv [S_i]P_i, \quad i \neq j, k$

W1 describes the actual placement of the dots. W2 is the result of the king's utterance: it is a common belief that at least one spot is white. W3 and W4 are schemata expressing the wise men's observational abilities, including the fact that everyone is aware of each other's capabilities. C1 is the circumscriptive ignorance axiom: the only beliefs a wise man has about the color of his own spot are the three axioms W2-W4, plus his observation of the other two wise men's spots.

As an exercise of the formalism, especially the circumscription rules, let us show that all agents are ignorant of the color of their own spot in the initial situation.

(5.2)

$$\begin{array}{c}
 \begin{array}{c}
 C_1 \quad \frac{C1 \Rightarrow \neg[S_i]P_i}{[S_i]P_i \supset \langle S_i : W2-4, P_j, P_k \rangle P_i \Rightarrow \neg[S_i]P_i} \\
 I_2 \quad \frac{CircK_1 \quad \frac{\langle S_i : W2-4, P_j, P_k \rangle P_i \Rightarrow \neg[S_i]P_i}{\not\models W2-4, P_j, P_k \Rightarrow_i P_i}}{N_1 \quad \frac{\Rightarrow [S_i]P_i, \neg[S_i]P_i}{[S_i]P_i \Rightarrow [S_i]P_i}} \\
 \times
 \end{array}
 \end{array}$$

We have omitted some irrelevant sentences from the left side of sequents in this tableau. To show that it closes, we must be able to prove that there is no set of sound deduction rules that will enable S_i to deduce P_i from W2, W3, W4, P_j , and P_k . We can prove this

for any set of sound tableau rules by showing that $W2-4, P_j, P_k \Rightarrow_i P_i$ is not provable in the normal modal logic $K4$ (see Section 4.4). It is possible to find a $K4$ -model in which the sequent $W2-4, P_j, P_k \Rightarrow_i P_i$ is false, using the methods of Sato [38]; hence this sequent is not provable in any logic of BK.

After the first wise man has spoken, it becomes a common belief that he does not know his own spot is white. The appropriate axioms are

$$(W5) \quad \neg[S_1]P_1 \wedge [S_0]\neg[S_1]P_1$$

$$(C2) \quad (S_i : W1-5, P_j, P_k)P_i \equiv [S_i]P_i, \quad i \neq j, k$$

In this new situation, all the wise men are again ignorant of their own spot's color; we could prove this fact, showing that $\vdash C2 \Rightarrow \neg[S_i]P_i$, in a manner similar to the proof in (5.2). S_2 relates his failure to the others, and the new situation has the additional axiom

$$(W6) \quad \neg[S_2]P_2 \wedge [S_0]\neg[S_2]P_2$$

The third wise man at this point does have sufficient cause to claim his spot is white, but only if the second wise man is indeed wise, and the third wise man believes he is. To see how this comes about, let us prove it in the saturated form of BK. We will take the wise men to be powerful reasoners, and set $r(\nu) = T_0 + AK^{CB} + CircK_1 + CircK_2$, for all views ν . The sequent we wish to prove is $W1-6 \Rightarrow [S_3]P_3$.

(5.3)

$$\begin{array}{c}
 I_2 \quad \frac{C_1 \quad \frac{W1-6 \Rightarrow [S_3]P_3}{W2-6, P_1, P_2, P_3 \Rightarrow [S_3]P_3}}{W2-6, P_1, P_2, P_3, P_2 \supset [S_3]P_2 \Rightarrow [S_3]P_3} \\
 \frac{P_2 \Rightarrow P_2 \quad \times \quad I_2 \quad \frac{C_1 \quad \frac{W2-6, P_1, P_2, P_3, [S_3]P_2 \Rightarrow [S_3]P_3}{W2-6, P_1, P_2, P_3, [S_3]P_2, P_1 \supset [S_3]P_1 \Rightarrow [S_3]P_3}}{P_1 \Rightarrow P_1 \quad \times \quad AK^{CB} \quad \frac{W2-6, P_1, P_2, P_3, [S_3]P_2, [S_3]P_1 \Rightarrow [S_3]P_3}{W2-6, P_1 \vee P_2 \vee P_3, P_2, P_1 \Rightarrow_3 P_3}}{P_3 \Rightarrow P_3}
 \end{array}$$

This part of the proof is mostly bookkeeping. We have used some shortcuts in the proof, omitting some obvious steps and dropping sentences from either side of the sequent if they are not going to be used.

We now must show that S_3 's belief subsystem can prove P_3 from the assumptions $W2-6$ and from the belief that the other two wise men's dots are white (note that we are now using S_3 's sequent \Rightarrow_3).

(5.4)

$$\begin{array}{c}
 I_2 \quad \frac{C_1 \quad \frac{W2-6, P_1 \vee P_2 \vee P_3, P_2, P_1 \Rightarrow_3 P_3}{W2-6, P_1, P_2, P_1 \supset [S_2]P_1 \Rightarrow_3 P_3}}{P_1 \Rightarrow_3 P_1} \\
 \times \\
 I_2 \quad \frac{C_1 \quad \frac{W2-6, P_1, P_2, [S_2]P_1 \Rightarrow_3 P_3}{W2-6, P_1, P_2, [S_2]P_1, \neg P_3 \supset [S_2]\neg P_3 \Rightarrow_3 P_3}}{\Rightarrow_3 P_3, \neg P_3} \\
 \times \\
 N_1 \quad \frac{P_3 \Rightarrow_3 P_3}{P_3 \Rightarrow_3 P_3} \\
 \times \\
 N_2 \quad \frac{W2-6, P_1, P_2, [S_2]P_1, [S_2]\neg P_3 \Rightarrow_3 P_3}{W2-6, P_1, P_2, [S_2]P_1, [S_2]\neg P_3 \Rightarrow_3 P_3, [S_2]P_2} \\
 AK \quad CB \\
 \frac{W2-6, P_1 \vee P_2 \vee P_3, P_1, \neg P_3 \Rightarrow_{32} P_2}{W2-6, P_1 \vee P_2 \vee P_3, P_1, \neg P_3 \Rightarrow_{32} P_2}
 \end{array}$$

Note the atom P_3 on the right-hand side of the top sequent; it is equivalent to $\neg P_3$ on the left-hand side, i.e., the assumption that S_3 's spot is black. The sequent proof here mimics the third wise man's reasoning, *Suppose my spot were black ...* Through the observation axiom $W4$, which is a common belief, this assumption means that S_3 believes that S_2 believes $\neg P_3$. At this point, S_3 begins to reason about S_2 's beliefs. Since, by $W6$, the second wise man is unaware of the color of his own spot, a contradiction will be derived if P_2 follows in S_2 's belief subsystem.

(5.5)

$$\begin{array}{c}
 I_2 \quad \frac{C_1 \quad \frac{W2-6, P_1 \vee P_2 \vee P_3, P_1, \neg P_3 \Rightarrow_{32} P_2}{W2-6, P_1, \neg P_3, \neg P_3 \supset [S_1]\neg P_3 \Rightarrow_{32} P_2}}{\neg P_3 \Rightarrow_{32} \neg P_3} \\
 \times \\
 I_2 \quad \frac{C_1 \quad \frac{W2-6, P_1, \neg P_3, [S_1]\neg P_3 \Rightarrow_{32} P_2}{W2-6, P_1, \neg P_3, [S_1]\neg P_3, \neg P_2 \supset [S_1]\neg P_2 \Rightarrow_{32} P_2}}{\Rightarrow_{32} P_2, \neg P_2} \\
 \times \\
 N_1 \quad \frac{P_2 \Rightarrow_{32} P_2}{P_2 \Rightarrow_{32} P_2} \\
 \times \\
 N_2 \quad \frac{W2-6, P_1, \neg P_3, [S_1]\neg P_3, [S_1]\neg P_2 \Rightarrow_{32} P_2}{W2-6, P_1, \neg P_3, [S_1]\neg P_3, [S_1]\neg P_2 \Rightarrow_{32} P_2, [S_1]P_1, [S_1]\neg P_1} \\
 AK \quad CB \\
 \frac{W2-6, P_1 \vee P_2 \vee P_3, \neg P_2, \neg P_3 \Rightarrow_{321} P_1}{W2-6, P_1 \vee P_2 \vee P_3, \neg P_2, \neg P_3 \Rightarrow_{321} P_1}
 \end{array}$$

S_2 's reasoning (in S_3 's view) takes the assumption that the third wise man's spot is black and asks what the effect would be on the first wise man S_1 . Since S_1 is also ignorant of the color of his own spot, a contradiction will ensue if the first wise man can prove that his own spot is white, under the assumption $\neg P_3$. The remainder of the proof is conducted in the view 321.

(5.6)

$$D_2 \quad \frac{N_2 \quad \frac{W2-6, P_1 \vee P_2 \vee P_3, \neg P_2, \neg P_3 \Rightarrow_{321} P_1}{W2-6, P_1 \vee P_2 \vee P_3, \Rightarrow_{321} P_1, P_2, P_3}}{P_1 \Rightarrow_{321} P_1, P_2, P_3 \quad P_2 \Rightarrow_{321} P_1, P_2, P_3 \quad P_3 \Rightarrow_{321} P_1, P_2, P_3}$$

In pursuing this proof, we have assumed that the second wise man is indeed wise. There are several places in which, with slightly less powerful deduction rules for the view 32, the proof would break down. Each of these corresponds to one of the two types of incompleteness that we identified in the statement of the problem: relevance incompleteness and fundamental logical incompleteness.

Consider first the notion that S_2 is not particularly good at reasoning about what other agents do not believe, a case of fundamental logical incompleteness. One way to capture this would be to weaken the rule N_2 in the following manner:

$$N'_2: \frac{\Gamma, \neg p \Rightarrow_{32} \Delta}{\Gamma \Rightarrow_{32} p, \Delta}, \quad \text{where } p \text{ contains no belief operators}$$

The modified rule N_2' would not allow deductions about what agents do not know. In particular, it would not allow the transfer of the sentence $\neg[S_1]P_1$ to the left-hand side of the sequent, a crucial step in the tableau (5.5) for the view \Rightarrow_{32} .

Note that the modified rule N'_2 still allows deductions about what other agents do believe. For instance, if S_2 were asked whether S_1 's believing P_1 followed from his believing $\neg P_2$ and $\neg P_3$, S_2 would say "yes," even with the logically incomplete rule N'_2 (as in tableau (5.6) above).

A more drastic case of logical incompleteness would result if S_2 simply did not reason about the beliefs of other agents at all. In that case, one would exclude the rule AK^CB from S_2 's deduction structure. Again, the proof would not go through, because the attachment rule could not be applied in the tableau (5.5).

The notion of relevance incompleteness emerges if the not-so-wise-man S_2 does not consider all the information he has available to answer the king. For example, he may

think that the observations of other agents are not relevant to the determination of his own spot, since the results of those observations are not directly available to him. The observational axioms $W3$ and $W4$ enter into the proof tableau (5.5) in two places. Both times the rule I_2 is used to break statements of the form $p \supset [S]p$ into their component atoms. Preventing the decomposition of $W3$ and $W4$ effectively prevents S_2 from reasoning about the observations of other agents. A weakened version of I_2 for doing this is:

$$I'_2: \frac{\Gamma, p \supset q \Rightarrow_{32} \Delta}{\Gamma \Rightarrow_{32} p, \Delta} \quad \Gamma, q \Rightarrow_{32} \Delta, \quad \text{where } p \text{ and } q \text{ are both modal or both nonmodal.}$$

This rule is actually weaker than required for the purpose we have in mind. Consider the observation axiom $\neg P_3 \supset [S_1]\neg P_3$. There are two ways S_2 could use this axiom. If S_2 believes $\neg P_3$, he could conclude that S_1 does also. This is not the type of deduction we wish to prevent, since it means that S_2 attributes beliefs to other agents based on his own beliefs about the world. On the other hand, the axiom $\neg P_2 \supset [S_1]\neg P_2$ is used in a conceptually different fashion. Here it is the contrapositive implication: if S_1 actually does not believe $\neg P_2$, then P_2 must hold. The way this shows up in the proof tableau (5.5) is that $\neg P_3$ appears as an initial assumption on the sequent $W2-5, P_1, \neg P_3 \Rightarrow_{32} P_2$, while P_2 is a goal to be proved.

To capture the notion of using an implicational sentence in one direction only, we would have to complicate the deduction rules by introducing asymmetry between the left and right sides of the sequent. This is one of the major strategies used by commonsense theorem provers of the PLANNER tradition (Hewitt [12] originated this theorem-proving method). Rather than having implicational rules of the form I_2 , typical PLANNER-type systems use something like the following rule.

$$PI: \frac{\Gamma, p, p \supset q \Rightarrow \Delta}{\Gamma, p, q, p \supset q \Rightarrow \Delta}$$

The implicational sentence is used in one direction only in PI . If it is desired to make contrapositive inferences, then the contrapositive form of the implication must be included explicitly. The construction of PLANNER-type deduction rules within the tableau framework allows a much finer degree of control over the inference process. A full exposition of such a system is beyond the scope of this paper; the interested reader is referred to Konolige [21].

In sum, we have shown that it is possible for the deduction model to represent the situation in which not-so-wise-man has less than perfect reasoning ability, preventing the third wise man from figuring out the color of his own spot. Both relevance incompleteness and fundamental logical incompleteness can be captured by using appropriate rules for $r(32)$.

8. Other Formal Approaches to Belief

How does the deduction model and its logic B compare to other formal models and logics of belief? We examine two alternative approaches in this section: modal logics based on a Hintikka/Kripke possible-worlds semantics, and several different first-order formalizations that treat beliefs as sentences in an internal language.

8.1. The Possible-Worlds Model

The possible-worlds model of belief was initially developed by Hintikka in terms of sets of sentences he called *model sets*. Subsequent to Kripke's introduction of possible worlds as a uniform semantics for various modal systems, Hintikka rephrased his work in these terms (see Hintikka [14]). The basic idea behind this approach is that the beliefs of an agent are modeled as a set of possible worlds, namely, those that are *compatible with* his beliefs. For example, an agent who believes the sentences

- (6.1) *Some of the artists are beekeepers.*
 All of the beekeepers are chemists.

would have his beliefs represented as the set of possible worlds in which some artists are beekeepers and all beekeepers are chemists.

Representational Issues

In a possible world for which the sentences (6.1) are true, anything that is a valid consequence of (6.1) must also be true. There can be no possible world in which some artists are beekeepers, all beekeepers are chemists, and no artists are chemists; such a world is a logical impossibility. If beliefs are compatible with a set of possible worlds (*i.e.*, true of each such possible world), then every valid consequence of those beliefs is also compatible

with the set. Thus one of the properties of the possible-worlds model is that an agent will believe all consequences of his beliefs – the model is consequentially closed. Hintikka, recognizing this as a serious shortcoming of the model, claimed only that it represented an idealized condition: an agent could justifiably believe any of the consequences of his beliefs, although in any given situation he might have only enough cognitive resources to derive a subset of them.

The assumption of consequential closure limits the ability of the possible-worlds model to represent the cognitive state of agents. Consider, for example, the problem of representing the mental state of agents as described by belief reports in a natural language. Suppose the state of John's beliefs is at least partially given by the sentence

- (6.2) *John believes that given the rules of chess, White has a forced initial win.*

Since the statement, *given the rules of chess, White has a forced initial win* is either a tautology or inconsistent, this would be equivalent in the possible-world model to one of the following belief reports:

- (6.3) a. *John believes t.*
 b. *John believes everything.*

Clearly this is wrong; if it turns out that John's belief in White's forced initial win is correct, John has a good deal of information about chess, and we would not want to equate it to the tautology *t*. On the other hand, if John's belief is false and no such strategy for White exists, it is not necessarily the case that all of his beliefs about other aspects of the world are incoherent. Yet there are no possible worlds compatible with a false belief, and so every proposition about the world must be a belief.

The representational problems of the possible-worlds approach stem from its treatment of belief as a relation between an agent and a proposition (*i.e.*, a set of possible worlds). All logically equivalent ways of stating the same proposition, no matter how complicated, count as a report of the same belief. By contrast, the deduction model treats belief as a relation between an agent and the *statement* of a proposition, so that two functionally different beliefs can have the same propositional content.

There is a large philosophical literature on the problems of representing propositional attitudes using possible worlds. Perry (in [34]) gives an account of some of the more subtle

problems inherent in equating belief states with propositions; his analysis does not depend on consequential closure. Barwise (in [2]) critiques consequential closure in possible-worlds models of perception. By comparison, a good account of the relative advantages of a symbol-processing approach to representing belief can be found in Moore and Hendrix (in [31]).

The Correspondence Property

It is reasonable to ask how the deduction and possible-worlds models compare in respects other than the assumption of consequential closure. That is, are the saturated deduction models $D_s(L, \rho)$ (whose rules are consequentially complete) significantly different from possible-worlds models for the purpose of representing belief?

The last phrase, "for the purpose of representing belief," is important. The two models are composed of different entities (expressions vs. propositions), so we can always use a language that distinguishes these entities, and has statements that are valid in one model and not the other. So the answer to this question depends on the type of language used to talk about the models. Fortunately, the language standardly used to axiomatize possible-worlds models is the same as that of B: a modal calculus containing atoms of the form $[S]p$, in which p refers to a proposition.¹ Thus it is possible to compare the possible-worlds and deduction models by comparing their axiomatizations in modal logic. We have proven the following general property about the two approaches.

Correspondence Property. For every modal logic of belief based on Kripke possible-worlds models, there exists a corresponding deduction model logic family with an equivalent saturated logic.

The correspondence property simply says that possible-worlds models are indistinguishable from saturated deduction models from the point of view of modal logics of belief. To the author's knowledge, this is the first time that the symbol-processing and possible-worlds approaches to belief have been shown to be comparable, in that the possible-worlds model

¹ Historically, the axiomatization of modal systems preceded Kripke's introduction of a unifying possible-worlds semantics.

is equivalent to the limiting case of a symbol-processing model with logically complete deduction.

Although space is too short here to give a full proof of this claim, we will give an overview of the most important of the propositional modal logics with a possible-worlds semantics, and their corresponding deductive belief logics (a full exposition and proofs of results mentioned here are in Konolige [21]).

Modal calculi for the possible-worlds model differ, depending on the particulars of their intended domains. For propositional modal calculi, these particulars center around whether knowledge or belief is being axiomatized, and what assumptions are made about self-beliefs or self-knowledge (a survey of these calculi may be found in Hughes and Cresswell [15]). The standard propositional modal calculi contain a single modal operator (which we write here as $[S]$) and are expressed as Hilbert systems. Their rules of inference are modus ponens (from p and $p \supset q$, infer q) and necessitation (from p , infer $[S]p$). Axioms are taken from the following schemata.

- M1. p , where p is a tautology
- M2. $[S](p \supset q) \supset ([S]p \supset [S]q)$
- M3. $[S]p \supset p$
- M4. $[S]p \supset [S][S]p$
- M5. $\neg[S]p \supset [S]\neg[S]p$

M1 are the purely propositional axioms. M2, also called the *distribution axioms*, allow modus ponens to operate under the scope of the modal operator. M3 are axioms for knowledge: all knowledge is true. M4 and M5 are called the positive and negative introspection axioms, respectively: if an agent believes p , then he believes that he believes it (M4); if he doesn't believe p , then he believes that he doesn't believe it (M5).

Any modal calculus that uses modus ponens and necessitation, and includes all tautologies and the distribution axioms, is called a *normal modal calculus*. Normal modal calculi have the following interesting property (see Boolos [3]): if $p \supset q$ is a theorem, then so is $[S]p \supset [S]q$. Interpreting the modal operator $[S]$ as belief, this asserts that whenever

q is implied by p , an agent S who believes p will also believe q . As expected, normal modal calculi assume consequential closure when the modal operator is interpreted as belief.

The simplest normal modal calculus is K , which contains just the schemata $M1$ and $M2$. To axiomatize knowledge, $M3$ is included to form the calculus T . Assumptions about self-knowledge lead to the calculi $S4$ ($T + M4$) and $S5$ ($S4 + M5$). McCarthy (in [24] and [25]) was the first to recognize the utility of modal calculi for reasoning about knowledge in AI systems, and defined three calculi that were extensions to T , $S4$, and $S5$, allowing belief operators for multiple agents. Sato ([38]) has a detailed analysis of these calculi as Gentzen systems, and calls them $K3$, $K4$, and $K5$, respectively. He also gives decision procedures for these logics. $K4$ is the calculus used by Moore in his dissertation on the interaction of knowledge and action ([29]).

The so-called weak analogs to $S4$ and $S5$ are formed by omitting the knowledge axiom $M3$ (this terminology is introduced by Stalnaker [41]). The weak versions are appropriate for axiomatizing belief rather than knowledge, since beliefs can be false. Levesque [22] has an interesting dissertation in which he explores the question of what knowledge a data base can have about its own information. Because he makes the assumption that a data base has complete and accurate knowledge of its own contents, the propositional calculus he arrives at is weak $S5$, with the addition of a consistency schema $[S]p \supset \neg[S]\neg p$.

How does the family of logics B compare with these propositional modal calculi? As with the possible-worlds logics, the deductive belief logics formed from B will depend on the assumptions that are made about self-beliefs. In this paper we have developed the logic family BK , which assumes that an agent has no knowledge of his own beliefs. The saturated logic BK_s , restricted to a single agent, is provably equivalent to K , the weakest of the possible-worlds belief calculi.

We have developed a theory of introspection within the deduction model framework that accounts for varying degrees of self-knowledge about one's own beliefs. This theory is based on the idea that an agent's belief subsystem can query a model of itself (an *introspective belief subsystem*) to answer question of self-belief. Depending on constraints placed on the introspective belief subsystem, it is possible to arrive at any one of eight

different logic families. Two of these, BS4 and BS5, have saturated logics that are equivalent in the single-agent case to the modal systems weak *S*4 and weak *S*5.

While we have been interested in the concept of belief throughout this paper, it is possible to define a deductive belief logic based on the related concept of knowledge. One property that distinguishes knowledge from belief is that if something is known it must be true, whereas beliefs can be false. The appropriate tableau axiom for knowledge is

$$K_0 : \frac{\Sigma, [S_i]\Gamma \Rightarrow \Delta}{\Sigma, \Gamma, [S_i]\Gamma \Rightarrow \Delta}$$

Adding K_0 to *B* forms the logic family *K*. Particularizations of *K* with varying degrees of self-knowledge correspond to the propositional modal systems *T*, *S*4 and *S*5.

We summarize these results in the following table.

	Normal Modal Calculus	Deduction Model Family
Belief	<i>K</i>	BK
	weak <i>S</i> 4	BS4
	weak <i>S</i> 5	BS5
Knowledge	<i>T</i>	KT
	<i>S</i> 4	KS4
	<i>S</i> 5	KS5

8.2. Syntactic Logics for Belief

There are a number of first-order formalizations of belief or knowledge in the symbol-processing tradition that have been proposed for AI systems. We have labeled these "syntactic" logics because their common characteristic is to have terms whose intended meaning is an expression of some object language. The object language is either a formal language (e.g., another first-order language) or an internal mental language. The logic *B* is also a syntactic logic, although it uses a modal operator; the argument of the operator denotes a sentence in the internal language. We have chosen to use a modal language for *B* because

it has a relatively simple syntax compared to first-order formalizations. It is also less expressive, in that quantification over sentences of the object language is not allowed by the modal syntax.

McCarthy [26] has presented some incomplete work in which *individual concepts* are reified in a first-order logic. Exactly what these concepts are is left deliberately unclear, but in one interpretation they can be taken for the internal mental language of a symbol-processing cognitive framework. He shows how the use of such concepts can solve the standard representational problems of knowledge and belief, e.g., distinguishing between *de dicto* and *de re* references in belief sentences.

A system that takes seriously the idea that agent's beliefs can be modeled as the theory of some first-order language is proposed by Konolige [19]. A first-order metalanguage is used to axiomatize the provability relation of the object language. To account for nested beliefs, the agent's object language is itself viewed as a metalanguage for another object language, and so on, thereby creating a hierarchy of metalanguage/object language pairs. Perlis [33] presents a more psychologically oriented first-order theory that contains axioms about long- and short-term memory. The ontology is that of an internal mental language.

These axiomatic approaches are marred by one or both of two defects - the lack of a coherent formal model of belief, and computational inefficiency. Regarding the first one: the vagueness of the intended model often makes it difficult to claim that the given axioms are the correct ones, since there is no formal mathematical model that is being axiomatized. In arriving at the deduction model of belief, we have tried to be very clear about what assumptions were being made in abstracting the model, how the model could fail to portray belief subsystems accurately, and so on. In contrast, the restrictions these syntactic systems place on belief subsystems are often obscure. What type of reasoning processes operate to produce consequences of beliefs? How are these processes invoked? What is the interaction of the belief subsystem with other parts of the cognitive model? These types of questions are begged when one simply writes first-order axioms and then tries to convey an intuitive idea of their intended content. (To some extent this criticism is not applicable to the formalism of Konolige in [19], because here the intended belief model is explicitly stated to be a first-order theory).

A second shortcoming is that efficient means of deduction for the syntactic axiomatizations are not provided. As we have mentioned, a system that is actually going to reason about belief by manipulating some formalization can encounter severe computational problems. Many of the assumptions incorporated into the deduction model, especially the closure property, were made with an eye towards deductive efficiency. The end result is a simple rule of inference, the attachment rule *A*, that has computationally attractive realizations.¹ On the other hand, formalizations that try to account for complex procedural interactions (as in Perlis's theory of long- and short-term memory), or that use a metalanguage to simulate a proof procedure at the object language level (as in Konolige [19]), have no obvious computationally efficient implementation.

¹ Several efficient proof methods are given in Konolige [21]: a decision procedure for propositional BK based on the Davis-Putnam procedure (see Chang and Lee [5]), which is sufficient to solve the Wise Man Puzzle automatically; a resolution method for the quantifying-in form of B; and a PLANNER-type deduction system.

7. Conclusion

We have explored a formalization of the symbol-processing paradigm of belief that we call the deduction model. It is interesting that the methodology employed was to examine the cognitive structure of AI planning systems. This methodology, which we might term *experimental robot psychology*, offers some distinct advantages over its human counterpart. Because the abstract design of such systems is open and available, it is possible to identify major cognitive structures, such as the belief subsystem, that influence behavior. Moreover, these structures are likely to be of the simplest sort necessary to accomplish some task, without the synergistic complexity so frequently encountered in studies of human intelligence. The design of a robot's belief subsystem is based on the minimum of assumptions necessary to ensure its ability to reason about its environment in a productive manner, namely, it incorporates a set of logical sentences about the world, and a theorem-proving process for deriving consequences. The deduction model is derived directly from these assumptions.

The deduction model falls within that finely bounded region between formally tractable but oversimplified models and more realistic but less easily axiomatized views. On the one hand, it is a generalization of the formal possible-worlds model that does not make the assumption of consequential closure, and so embodies the notion that reasoning about one's beliefs is resource-limited. On the other hand, it possesses a concise axiomatization in which an agent's belief deduction process is incorporated in a direct manner, rather than simulated indirectly. Thus, the deduction model and its associated logic B lend themselves to implementation in mechanical theorem-proving processes as a means of giving AI systems the capability of reasoning about beliefs.

Acknowledgements

Many people contributed their time and effort to reading and critiquing this paper. I am especially indebted to Stan Rosenschein, Nils Nilsson, and Jerry Hobbs in this regard.

References

- [1] Appelt, D. E., "Planning Natural-Language Utterances to Satisfy Multiple Goals," *SRI Artificial Intelligence Center Technical Note 259*, SRI International, Menlo Park, California (1982).
- [2] Barwise, J., "Scenes and other Situations," *Journal of Philosophy* LXXVIII, 7 (July, 1981).
- [3] Boolos, G., *The Unprovability of Consistency*, Cambridge University Press, Cambridge, Massachusetts, 1979.
- [4] Brachman, R., "Recent Advances in Representational Languages," Invited lecture at the National Conference on Artificial Intelligence, Stanford University, Stanford, California (1980).
- [5] Chang, C. L. and Lee, R. C. T., *Symbolic Logic and Mechanical Theorem Proving*, Academic Press, New York, New York, 1973.
- [6] Collins, A. M., Warnock, E., Aiello, N. and Miller, M., "Reasoning from Incomplete Knowledge," in *Representation and Understanding*, Bobrow, D. G., and Collins, A. (eds.), Academic Press, New York (1975).
- [7] Doyle, J., "Truth Maintenance Systems for Problem Solving," *Artificial Intelligence Laboratory Technical Report 419*, Massachusetts Institute of Technology, Cambridge, Massachusetts (1978).
- [8] Field, H. H., "Mental Representation," *Erkenntnis* 13 (1978), pp. 9-61.
- [9] Fikes, R. E. and Nilsson, N. J., "STRIPS: A New Approach to the Application of Theorem Proving to Problem Solving," *Artificial Intelligence* 2, 3-4 (1971).
- [10] Fodor, J. A., *The Language of Thought*, Thomas Y. Cromwell Company, New York, New York, 1975.
- [11] Hayes, P. J., "In Defence of Logic," *Proceedings of the 5th International Joint Conference on Artificial Intelligence*, Massachusetts Institute of Technology, Cambridge, Massachusetts (1977).
- [12] Hewitt, C., *Description and Theoretical Analysis (Using Schemata) of PLANNER: A Language for Proving Theorems and Manipulating Models in a Robot*, Doctoral dissertation, Massachusetts Institute of Technology, Cambridge, Massachusetts, 1972.

- [13] Hintikka, J., "Form and Content in Quantification Theory," *Acta Philosophica Fennica* 8 (1955), pp. 7-55.
- [14] Hintikka, J., *Knowledge and Belief*, Cornell University Press, Ithaca, New York, 1962.
- [15] Hughes, G. E. and Cresswell, M. J., *Introduction to Modal Logic*, Methuen and Company Ltd., London, England, 1968.
- [16] Israel, D. J., "The Role of Logic in Knowledge Representation," *Computer* 18, 10 (October, 1983).
- [17] Johnson-Laird, P. N., "Mental Models in Cognitive Science," *Cognitive Science* 4 (1980), pp. 71-115.
- [18] Kleene, S. C., *Mathematical Logic*, John Wiley and Sons, New York, 1967.
- [19] Konolige, K., "A First Order Formalization of Knowledge and Action for a Multi-agent Planning System," in *Machine Intelligence 10*, J. E. Hayes, D. Michie, and Y-H Pao (eds.), Ellis Horwood Limited, Chichester, England (1982).
- [20] Konolige, K., "Circumscriptive Ignorance," *Proceedings of the Second National Conference on Artificial Intelligence*, Carnegie-Mellon University, Pittsburgh, Pennsylvania (1982).
- [21] Konolige, K., *A Deduction Model of Belief and its Logics*, Doctoral thesis in preparation, Stanford University Computer Science Department, Stanford, California, 1984.
- [22] Levesque, H. J., "A Formal Treatment of Incomplete Knowledge Bases," *FLAIR Technical Report No. 614*, Fairchild, Palo Alto, California (1982).
- [23] Lycan, W. G., "Toward a Homuncular Theory of Believing," *Cognition and Brain Theory* 4, 2 (1981), pp. 139-59.
- [24] McCarthy, J., Sato, M., Hayashi, T., and Igarashi, S., "On the Model Theory of Knowledge," *Stanford Artificial Intelligence Laboratory Memo AIM-312*, Stanford University, Stanford (1978).
- [25] McCarthy, J., "Formalization of two puzzles involving knowledge," unpublished note, Stanford University, Stanford, California (1978).
- [26] McCarthy, J., "First Order Theories of Individual Concepts and Propositions," in *Machine Intelligence 9*, B. Meltzer and D. Michie (eds.), Edinburgh University Press, Edinburgh, England (1979), pp. 120-147.
- [27] McCarthy, J., "Circumscription-A Form of Non-Monotonic Reasoning," *Artificial Intelligence* 13, 1,2 (1980).
- [28] McDermott, D. and Doyle, J., "Non-Monotonic Logic I," *Artificial Intelligence* 13, 1,2 (1980).
- [29] Moore, R. C., "Reasoning About Knowledge and Action," *Artificial Intelligence Center Technical Note 191*, SRI International, Menlo Park, California (1980).

- [30] Moore, R. C., "Semantical Considerations on Nonmonotonic Logic," *SRI Artificial Intelligence Center Technical Note 284*, SRI International, Menlo Park, California (June, 1983).
- [31] Moore, R. C. and Hendrix, G. G., "Computational Models of Belief and the Semantics of Belief Sentences," *SRI Artificial Intelligence Center Technical Note 187*, SRI International, Menlo Park, California.
- [32] Nilsson, N., *Principles of Artificial Intelligence*, Tioga Publishing Co., Palo Alto, California, 1980.
- [33] Perlis, D., "Language, Computation, and Reality," *Department of Computer Science TR95*, University of Rochester, Rochester, New York (May, 1981).
- [34] Perry, J., "The Problem of the Essential Indexical," *NOÛS* 13 (1979).
- [35] Reiter, R., "A Logic for Default Reasoning," *Artificial Intelligence* 13, 1-2 (1980).
- [36] Robinson, J. A., *Logic: Form and Function*, Elsevier North Holland, New York, New York, 1979.
- [37] Sacerdoti, E. D., *A Structure for Plans and Behavior*, Elsevier, New York, 1977.
- [38] Sato, M., *A Study of Kripke-type Models for Some Modal Logics by Gentzen's Sequential Method*, Research Institute for Mathematical Sciences, Kyoto University, Kyoto, Japan, July 1976.
- [39] Schubert, L. K., "Extending the Expressive Power of Semantic Nets," *Artificial Intelligence* 7, 2 (1976), pp. 163-198.
- [40] Smullyan, R. M., *First-Order Logic*, Springer-Verlag, New York, 1968.
- [41] Stalnaker, R., "A Note on Nonmonotonic Modal Logic," unpublished manuscript, Department of Philosophy, Cornell University (1980).
- [42] Warren, D. H. D., "WARPLAN: A System for Generating Plans," *Dept. of Computational Logic Memo 76*, University of Edinburgh School of Artificial Intelligence, Edinburgh, England (1974).
- [43] Weyhrauch, R., "Prolegomena to a Theory of Mechanized Formal Reasoning," *Artificial Intelligence* 13 (1980).
- [44] Woods, W., "What's in a Link?," in *Representation and Understanding*, Bobrow, D. G., and Collins, A. (eds.), Academic Press, New York (1975).

Appendix E

POSSIBLE-WORLD SEMANTICS FOR AUTOEPISTEMIC LOGIC



POSSIBLE-WORLD SEMANTICS FOR AUTOEPISTEMIC LOGIC

Technical Note 337

August 1984

By: Robert C. Moore, Staff Scientist
Artificial Intelligence Center
Computer Science and Technology Division

SRI Project 4488

To be presented at the Workshop on Nonmonotonic Reasoning, Mohonk Mountain House, New Paltz, New York, October 17-19, 1984.

The research reported herein was supported by the Air Force Office of Scientific Research under Contract No. F49620-82-K-0031. The views and conclusions expressed in this document are those of the author and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the Air Force Office of Scientific Research or the U.S. Government.

ABSTRACT

In a previous paper [Moore, 1983a, 1983b], we presented a nonmonotonic logic for modeling the beliefs of ideally rational agents who reflect on their own beliefs, which we called "autoepistemic logic." We defined a simple and intuitive semantics for autoepistemic logic and proved the logic sound and complete with respect to that semantics. However, the nonconstructive character of both the logic and its semantics made it difficult to prove the existence of sets of beliefs satisfying all the constraints of autoepistemic logic. This note presents an alternative, possible-world semantics for autoepistemic logic that enables us to construct finite models for autoepistemic theories, as well as to demonstrate the existence of sound and complete autoepistemic theories based on given sets of premises.

I INTRODUCTION

In a previous paper [Moore, 1983a, 1983b], we presented a nonmonotonic logic for modeling the beliefs of ideally rational agents who reflect on their own beliefs, which we called "autoepistemic logic." We defined a simple and intuitive semantics for autoepistemic logic and proved the logic sound and complete with respect to that semantics. However, the nonconstructive character of both the logic and its semantics made it difficult to prove the existence of sets of beliefs satisfying all the constraints of autoepistemic logic. This note presents an alternative, possible-world semantics for autoepistemic logic that enables us to construct finite models for autoepistemic theories, as well as to demonstrate the existence of sound and complete autoepistemic theories based on given sets of premises.

Autoepistemic logic is nonmonotonic, because we can make statements in the logic that allow an agent to draw conclusions about the world from his own lack of information. For example, we can express the belief that "If I do not believe P, then Q is true." If an agent adopts this belief as a premise and he has no means of inferring P, he will be able to derive Q. On the other hand, if we add P to his premises, Q will no longer be derivable. Hence, the logic is nonmonotonic.

Autoepistemic logic is closely related to the nonmonotonic logics of McDermott and Doyle [1980; McDermott, 1982]. In fact, it was designed to be a reconstruction of these logics that avoids some of their peculiarities. This is discussed in detail in our earlier paper [Moore, 1983a, 1983b]. This work is also closely related to that of Halpern and Moses [1984], the chief difference being that theirs is a logic of knowledge rather than belief. Finally, Levesque [1981] has also developed a kind of autoepistemic logic, but in his system the agent's premises are restricted to a sublanguage that makes no reference to what he believes.

II SUMMARY OF AUTOEPISTEMIC LOGIC

The language of autoepistemic logic is that of ordinary propositional logic, augmented by a modal operator L . We want formulas of the form LP to receive the intuitive interpretation "P is believed" or "I believe P." For example, $P \supset LP$ could be interpreted as saying "If P is true, then I believe that P is true."

The type of object that is of primary interest in autoepistemic logic is a set of formulas that can be interpreted as a specification of the beliefs of an agent reflecting upon his own beliefs. We will call such a set of formulas an autoepistemic theory. The truth of an agent's beliefs, expressed as an autoepistemic theory, is determined by (1) which propositional constants are true in the external world and (2) which formulas are believed by the agent. A formula of the form LP will be true with respect to an agent if and only if P is in his set of beliefs. To formalize this, we define the notions of autoepistemic interpretation and autoepistemic model. An autoepistemic interpretation I of an autoepistemic theory T is a truth assignment to the formulas of the language of T that satisfies the following conditions:

1. I conforms to the usual truth recursion for propositional logic.
2. A formula LP is true in I if and only if $P \in T$.

An autoepistemic model of T is an autoepistemic interpretation of T in which all the formulas of T are true. (Any truth assignment satisfying Condition 1 in which all the formulas of T are true will be called simply a model of T.)

We can readily define notions of soundness and completeness relative to this semantics. Soundness of a theory must be defined with respect to some set of premises. Intuitively speaking, an autoepistemic theory T, viewed as a set of beliefs, will be sound with respect to a set of premises A, just in case every formula in T must be true, given that all the formulas in A are true and given that T is, in fact, the set of beliefs under consideration. This is expressed formally by the following definition:

An autoepistemic theory T is sound with respect to a set of premises A if and only if every autoepistemic interpretation of T that is a model of A is also a model of T.

The definition of completeness is equally simple. A semantically complete set of beliefs will be one that contains everything that must be true, given that the entire set of beliefs is true and given that it is the set of beliefs being reasoned about. Stated formally, this becomes

An autoepistemic theory T is semantically complete if and only if T contains every formula that is true in every autoepistemic model of T.

Finally, we can give syntactic characterizations of the autoepistemic theories that conform to these definitions of soundness and completeness [Moore, 1983b, Theorems 3 and 4]. We say that an autoepistemic theory T is stable if and only if (1) it is closed under ordinary tautological consequence, (2) $LP \in T$ whenever $P \in T$, and (3) $\neg LP \in T$ whenever $P \notin T$.

Theorem: An autoepistemic theory T is semantically complete if and only if T is stable.

We say that an autoepistemic theory T is grounded in a set of premises A if and only if every formula in T is a tautological consequence of $A \cup \{LP \mid P \in T\} \cup \{\neg LP \mid P \notin T\}$.

Theorem: An autoepistemic theory T is sound with respect to a set of premises A if and only if T is grounded in A .

With these soundness and completeness theorems, we can see that the possible sets of beliefs an ideally rational agent might hold, given A as his premises, would be stable autoepistemic theories that contain A and are grounded in A . We call these theories stable expansions of A .

III AN ALTERNATIVE SEMANTICS FOR AUTOEPISTEMIC LOGIC

The semantics we have provided for autoepistemic logic is simple, intuitive, and allows us to prove a number of important general results, but it requires enumerating an infinite truth assignment if the theory under consideration contains infinitely many formulas. This makes it difficult to exhibit particular models and interpretations we may be interested in. The problem is that, in the general case, there need be no systematic connection between the truth of one formula of the form LP and any other. Autoepistemic logic is designed to characterize the beliefs of ideally rational agents, but we want the semantics to be broader than that. The semantics we have defined is intended to apply to arbitrary sets of beliefs, with the beliefs of ideally rational agents being a special case (just as model theory for standard logic applies to arbitrary sets of formulas, not just to those that are closed under logical consequence). Thus, our semantics makes no necessary connection between the truth of $L(P \wedge Q)$ and LP or LQ , because it is at least conceivable that an agent might be so logically deficient as to believe $P \wedge Q$ without believing P or believing Q . In such a case, there is little we can expect the truth definition for an autoepistemic theory to do, other than to list the true formulas of the form LP by brute stipulation.

If we confine our attention to ideally rational agents, however, much more structure emerges. In fact, we can show that stable autoepistemic theories can be simply characterized by Kripke-style possible-world models for modal logic [Kripke, 1971]. For our purposes, what we need to recall about a Kripke structure is that it contains a set of possible worlds and an accessibility relation between pairs of worlds. The truth of a formula is defined relative to a world, and conforms to the usual truth recursion for propositional logic. A formula of the form LP is true in a world W just in case P is true in every world accessible from W . Kripke structures in which the accessibility relation is an equivalence relation are called S5 structures, and the S5 structures that will be of interest to us are those in which every world is accessible from every world. We will call these the complete S5 structures. Our major result is that the sets of formulas that are true in every world of some complete S5 structure are exactly the stable autoepistemic theories. (This result has been obtained independently by Halpern and Moses [1984] and by Melvin Fitting [personal communication]).

Theorem: T is the set of formulas that are true in every world of some complete S5 structure if and only if T is a stable autoepistemic theory.

Proof: Suppose T is the set of formulas true in every world of a complete S5 structure. By the soundness of propositional logic, T is closed under tautological consequence. By the truth rule for L , LP is true in every world just in case P is true in every world; therefore

$LP \in T$ if and only if $P \in T$. Furthermore, by the truth rule for L , LP is false in every world just in case P is false in some world; so $\neg LP \in T$ if and only if $P \notin T$. Therefore T is stable. In the opposite direction, suppose that T is stable. Let T' be the set of formulas of T that contain no occurrences of L . We will call these the objective formulas of T . Since T is closed under tautological consequence, T' will also be closed under tautological consequence. Consider the set of all models of T' and the complete S5 structure in which each of these models defines a possible world. T' will contain exactly the objective formulas true in every world in this model; hence, T' will contain precisely the objective formulas of the stable autoepistemic theory T'' defined by this S5 structure. But by a previous result [Moore 1983b, Theorem 2], stable theories containing the same objective formulas are identical, so T must be the same as T'' . Hence, T is the set of formulas true in every world of a complete S5 structure.

Given this result, we can characterize any autoepistemic interpretation of any stable theory by an ordered pair consisting of a complete S5 structure (to specify the agent's beliefs) and a propositional truth assignment (to specify what is actually true in the world). Such a structure (K, V) defines an autoepistemic interpretation of the theory T consisting of all the formulas that are true in every world in K . A formula of T is true in (K, V) if it is true according to the standard truth recursion for propositional logic, where the propositional constants are true in (K, V) if and only if they are true

in V , and the formulas of the form LP are true in (K, V) if and only if they are true in every world in K (using the truth rules for Kripke structures). We will say that (K, V) is a possible-world interpretation of T and, if every formula of T is true in (K, V) , we will say that (K, V) is also a possible-world model of T . In view of the preceding theorem, it should be obvious that for every autoepistemic interpretation or autoepistemic model of a stable theory there is a corresponding possible-world interpretation or possible-world model, and vice versa.

Theorem: If (K, V) is a possible-world interpretation of T , then (K, V) will be a possible-world model of T if and only if the truth assignment V is consistent with the truth assignment provided by one of the possible worlds in K (i.e., if the actual world is one of the worlds that are compatible with what the agent believes).

Proof: If V is compatible with one of the worlds in K , then any propositional constant that is true in all worlds in K will be true in V . Therefore, any formula that comes out true in all worlds in K will also come out true in (K, V) , and (K, V) will be a possible-world model of T . In the opposite direction, suppose that V is not compatible with any of the worlds in K . Then, for each world W in K , there will be some propositional constant that W and V disagree on. Take that constant or its negation, whichever is true in W , plus the corresponding formulas for all other worlds in K , and form their disjunction. (This will be a finite disjunction, provided there are only finitely many propositional constants in the language.) This disjunction will be true in every

world in K , so it will be a formula of T , but it will be false in V .
Therefore, (K, V) will not be a possible-world model of T .

IV APPLICATIONS OF POSSIBLE-WORLD SEMANTICS

One of the problems with our original presentation of autoepistemic logic was that, since both the logic and its semantics were defined nonconstructively, we were unable to easily prove the existence of stable expansions of nontrivial sets of premises. With the finite models provided by the possible-world semantics for autoepistemic logic, this becomes quite straightforward. For instance, we claimed [Moore, 1983a, 1983b] that the set of premises $\{\neg LP \supset Q, \neg LQ \supset P\}$ has two stable expansions--one containing P but not Q , and the other containing Q but not P --but we were unable to do more than give a plausibility argument for that assertion. We can now demonstrate this fact quite rigorously.

Consider the stable theory T , generated by the complete S5 structure that contains exactly two worlds, $\{P, Q\}$ and $\{P, \neg Q\}$. (We will represent a possible world by the set of propositional constants and negations of propositional constants that are true in it.) The possible-world interpretations of T will be the ordered pairs consisting of this S5 structure and any propositional truth assignment. Consider all the possible-world interpretations of T in which $\neg LP \supset Q$ and $\neg LQ \supset P$ are both true. By exhaustive enumeration, it is easy to see that these are exactly

$$(\{P, Q\}, \{P, \neg Q\}, \{P, Q\})$$

$$(\{P, Q\}, \{P, \neg Q\}, \{P, \neg Q\})$$

Since, in each case, the actual world is one of the worlds that are compatible with everything the agent believes, each of these is a possible-world model of T . Therefore, T is sound with respect to $\{\neg LP \supset Q, \neg LQ \supset P\}$. Since T is stable and includes $\{\neg LP \supset Q, \neg LQ \supset P\}$ (note that both these formulas are true in all worlds in the S5 structure), T is a stable expansion of A . Moreover, it is easy to see that T contains P but not Q . A similar construction yields a stable expansion of T that contains Q but not P .

On the other hand, if both P and Q are to be in a theory T , the corresponding S5 structure contains only one world, $\{P, Q\}$. But then $(\{P, Q\}, \{P, \neg Q\})$ is a possible-world interpretation of T in which $\neg LP \supset Q$ and $\neg LQ \supset P$ are both true, but some of the formulas of T are not (P and Q , for instance). Hence, if T contains both P and Q , T is not a stable expansion of $\{\neg LP \supset Q, \neg LQ \supset P\}$.

REFERENCES

- Kripke, S. A. [1971] "Semantical Considerations on Modal Logic," in Reference and Modality, L. Linsky, ed., pp. 63-72 (Oxford University Press, London, England).
- Halpern, J. Y. and Y. Moses [1984] "Towards a Theory of Knowledge and Ignorance," Workshop on Nonmonotonic Reasoning, Mohonk Mountain House, New Paltz, New York (October 17-19, 1984).
- Levesque, H. J. [1981] "The Interaction with Incomplete Knowledge Bases: A Formal Treatment," Proceedings of the Seventh International Joint Conference on Artificial Intelligence, University of British Columbia, Vancouver, B.C., Canada, pp. 240-245 (August 24-28, 1981).
- McDermott, D. and J. Doyle [1980] "Non-Monotonic Logic I," Artificial Intelligence, Vol. 13, Nos. 1, 2, pp. 41-72 (April 1980).
- McDermott, D. [1982] "Nonmonotonic Logic II: Nonmonotonic Modal Theories," Journal of the Association for Computing Machinery, Vol. 29, No. 1, pp. 33-57 (January 1982).
- Moore R. C. [1983a] "Semantical Considerations on Nonmonotonic Logic," Proceedings of the Eighth International Joint Conference on Artificial Intelligence, Karlsruhe, West Germany, pp. 272-279 (August 8-12, 1983).
- Moore R. C. [1983b] "Semantical Considerations on Nonmonotonic Logic," SRI Artificial Intelligence Center Technical Note 284, SRI International, Menlo Park, California (June 1983).

Appendix F

A WEAK LOGIC FOR KNOWLEDGE AND BELIEF



A WEAK LOGIC OF KNOWLEDGE AND BELIEF: Epistemic and Doxastic Logic for the Yuppie Generation

Technical Note 359

By: David Israel
Senior Computer Scientist

Artificial Intelligence Center
Computer Science and Technology Division

This research was supported in part by the United States Air Force
Office of Scientific Research under contract No. F49620-82-K-0031
and in part by a gift from the System Development Foundation

Table of Contents

1. INTRODUCTION	0
2. ON AXIOMATIZING KNOWLEDGE AND BELIEF	2
3. INTRODUCING SCOTT AND KIMBERLY	4
4. ON KNOWLEDGE.	5
5. ON BELIEF	8
6. ON LIMITING INTROSPECTION FOR BELIEF	10
6.1. <i>Model Theory of Intensional Logics</i>	10
6.2. <i>Some Applications.</i>	12
6.3. <i>A Few Doxastic Paradoxes</i>	13
6.4. <i>Moore's Paradox and the Schema Y</i>	15
6.5. <i>More on Y</i>	19
7. SUMMING UP	22
8. SOME FINAL SCEPTICAL REMARKS.	24
8.1. <i>On Belief States</i>	24
8.2. <i>On the Contents of Beliefs</i>	25

A WEAK LOGIC OF KNOWLEDGE AND BELIEF:

Epistemic and Doxastic Logic for the Yuppie Generation¹

David Israel
Artificial Intelligence Center
SRI International²

1. INTRODUCTION

Modern modal logic begins with the work of C. I. Lewis early on in the present century [Lewis 18]. We can think of Lewis thinking to himself as follows: "Well, I can't analyze the notions of metaphysical or logical possibility and necessity, but I can sure formulate alternative axiomatizations of such notions. I can then compare and contrast such axiomatic systems and see what I learn." Thus were born the Lewis Systems, S1-S5, axiomatizing increasingly strong conceptions of necessity.³

Another 40 or so years went by before the purely axiomatic approach was properly systematized and rendered fit for human consumption. In current lore, a certain axiomatic system, *K*, is central.⁴ The standard presentation of *K* consists of infinitely many axioms plus one axiom scheme and two rules of inference. In particular, with 'L' being read as "necessarily" or "it is necessary that"; 'M', as "possibly" or "it is possible that", *K* is as follows:

I: all classical tautologies

II: $L(p \rightarrow q) \rightarrow (Lp \rightarrow Lq)$

¹I will now show off almost all the Greek I know: "epistemic" has to do with knowledge; "doxastic", with belief. So in what follows we shall have to do with logics of knowledge and belief.

²This research was supported in part by the United States Air Force Office of Scientific Research under Contract No. F49620-82-K-0031 and in part by a gift from the System Development Foundation.

³The little story just told is a fable. Lewis was really interested in different conceptions of implication or the conditional—not in varying conceptions of necessity and possibility. Of course, on one view, implication simply *is* validity or *necessity* of the material conditional; so we can translate Lewis's writings on the varieties of implication into writings on varieties of necessity. This translation scheme is now almost universally applied. Note, if one does not apply this scheme, and instead reads Lewis neat, the proper line of descent from Lewis goes mainly through Ackermann's work on 'strange Implication' to the work of Anderson-Belnap on entailment. See [Anderson and Belnap 75].

⁴The "K" is for Kripke, although credit for focussing on a notion of normality under which *K* is the minimal normal modal logic must be shared with E.J. Lemmon [Lemmon 77]. See below on normality.

R1: If $\Box p$ and $\Box(p \rightarrow q)$, then $\Box q$ *modus ponens*

R2: If $\Box p$, then $\Box \Box p$ *necessitation*

The standard practice is to take K as the base theory and consider extensions. Four such extensions have figured prominently in the literature.

$$T: K + \Box p \rightarrow p$$

$$S4: T + \Box p \rightarrow \Box \Box p$$

$$B: T + \Box \Box p \rightarrow p$$

$$S5: T + \Box \Box p \rightarrow \Box p$$

In all of these logics, possibility and necessity are duals; that is, in all of them " $\Box p$ " is provably equivalent to " $\neg \Box \neg p$ " and " $\Box p$ " to " $\Box \Box p$ ". Thus they can all be with only one primitive modal operator (\Box or \Diamond)--its dual (\Diamond or \Box , respectively) being introduced by definitional abbreviation.

Just to confuse the reader, I shall spend a little time on alternative systems of nomenclature for modal systems. First, and least annoying, T is also referred to as M . Now then, look at the characterization of, say, M . (Just testing.) M is presented as K plus one axiom schema. That schema is also often referred to as **T**--though never, I think, as **M**. Thus T , the system, just is $K + \mathbf{T}$, the schema. This particular annoyance, or variants of it, recurs. The schema, which when added to $K + \mathbf{T}$ yields $S4$, is called **4**; that, which when added to $K + \mathbf{T}$ yields B , is **B**. Finally, the $S5$ schema is **E**. The scorecard looks like this:

$$T = K + \mathbf{T}$$

$$S4 = K + \mathbf{T} + \mathbf{4}$$

$$B = K + \mathbf{T} + \mathbf{B}$$

$$S5 = K + \mathbf{T} + \mathbf{E}$$

In the remainder of this paper, I shall adhere to the conventions manifested on the right hand side of these equations; thus, I shall be looking at systems that are presented as $K + \mathbf{X}$, \mathbf{X} the unknown.

2. ON AXIOMATIZING KNOWLEDGE AND BELIEF

To return to the main line: these four standard modal logics were meant to formalize different conceptions of necessity and possibility. They were not meant to cast any light on the notions of knowledge or belief--or on different conceptions of knowledge or belief. Indeed, what *a priori* reason is there to believe that any of these standard logics of **necessity** are appropriate logics of **knowledge** or **belief**? Whatever the answer to that, Hintikka [Hintikka 62] gave people lots of reasons *a posteriori* to think that (1) $K + 4$ was an appropriate logic for knowledge and (2) $K + E$ was an appropriate logic for belief. (Note: $K + E = S5 - T$. This is sometimes called "weak S5.")⁵

The response to Hintikka's work was quite stunning--as these things go; and as they went, no one paid much mind to the logic of belief. The focus was squarely on knowledge--to philosophers, at any rate, the more interesting and more discussed notion. Many attempts at conceptual analysis of the notion of knowledge had been made; none had met with exactly universal acceptance. So why not go Lewis's route: don't analyze, axiomatize? **Especially now!!**

Why especially now?? Because in the interim (1918 to 1962), logicians had come up with model-theoretic tools for a variety of modal logics--including our four standard ones. (It was a number of years before it was clear how wide a variety this was.) Further on, we shall look at the main ingredients of the now standard model theoretic treatment; for now, it suffices to note its very existence and to note that its existence played a large part in the excitement surrounding Hintikka's work.⁶

Still, there was trouble in the new paradise. It came in two quite independent forms. First, there was the problem of *logical omniscience*, so-called. Then, there were problems about introspection. As for the first problem: it is easy to prove that K by itself--with 'K' substituted for 'L', of course--guarantees both that every classical tautology is known and that knowledge is closed under classical tautological consequence. The latter means that if S' follows tautologously from S and if it is known that S , then it is

⁵The sharp-eyed reader might have guessed that there were more notational headaches ahead. However it came to be that 'L' got associated with "it is necessary that" and 'M' with "it is possible that". It was only to be expected that 'K' would be used for "it is known that" and 'B' for "it is believed that". But now 'K' stands for both an axiomatic system and a modal operator; 'B', for a modal system, an axiom schema, *and* a modal operator. Context, together with my convention of *italicizing* system names and **boldfacing** schema names, will disambiguate. By the way, I trust that it is clear that knowledge ('K') and belief ('B') are not duals. From "it is not believed that it is not the case that p ", "it is known that p " does not follow; nor vice versa. Nor should one infer from "it is not the case that it is known that it is not the case that p " to "it is believed that p "; or vice versa.

⁶More fabulating; Hintikka's original work was not done within the then new model theoretic framework; the "semantic" machinery was, rather, syntactic and proof-theoretic. In later versions, Hintikka did adopt the new standard.

known that S .⁷ Idealization is fine, indeed necessary in any science; but surely this is going too far with a fine thing.

The second set of problems had to do with what one should add to $K + T$ for knowledge or to plain old K for belief. (Remember that, sad to say, we can't allow ourselves T for belief.) Hintikka spends a good deal of time arguing for the inclusion of **4**, at least for knowledge. Many thought that this was too strong a requirement. He also argued against the inclusion, again for knowledge, of **B** and **E**. Here the consensus was with him. Questions were raised about belief as well. Could one believe that p without believing that one believed that p ? That is, should one add **4** to K ? Could one not believe that p without believing that one did not believe that p ? That is, should one add **E** to K ?⁸

⁷ A guarantee: more if S is any theorem of K --it need not be a classical tautology-- then it is known that S ; this is just what the rule of necessitation yields. Mutatis mutandis for closure under consequence; think of it as closure under K -consequence.

⁸ A word in explanation of the grotesqueries of logician's English. "Scott doesn't believe that p " is ambiguous. It can be understood to mean that Scott--for whom, see below--believes that not- p or to mean simply that it is not the case that he believes that p . Scott might not have any fixed opinion as to whether p . In what follows, it is crucial that these two readings be distinguished; the ugly way, deploying negation only as a sentence-level operator in the guise "it is not the case that", is the way for me. To make matters worse, I refuse to countenance any natural dual for either "knows" or "believes", either ' K ' or ' M '. It is nice that "necessarily" and "possibly" are (arguably) lexicalized duals; thus, we don't have to keep writing down things like "it is not the case that it is necessary that it is not the case that...". We can write instead "it is possible that...". But not only aren't "knows" and "believes" duals, neither has a natural, lexical dual. So there will be lots of ugly things like "it is not the case that Scott believes that it is not the case that Scott believes that Scott believes that p ". Sorry.

3. INTRODUCING SCOTT AND KIMBERLY

To fix ideas, let's imagine a subject. To fix our perhaps sexist imaginations, let's imagine two subjects, Scott and Kimberly. So, in what follows 'K' is to be read as "Scott (Kimberly) knows that..." and 'B', as "Kimberly (Scott) believes that..." The formalisms I will be discussing are all of the single subject variety. I shall have nothing to say about the multisubject versions being studied by researchers in theoretical computer science interested in distributed systems [Halpern and Moses 84].⁹

Scott and Kimberly are, of course, terrifically bright; but are they logically omniscient? Why not make their mommies and daddies happy by assuming that they are. This decision also makes me happy: for a mixture of tactical and technical reasons, I think it useful to retain *K* as our base theory. For alternatives to this, see [Fagin and Halpern 85].

In any case, unrestricted necessitation is out for any applied epistemic or doxastic logic. Imagine that we are interested in some set of putative facts and in what Kimberly knows/believes about them. One such fact might be that South San Francisco calls itself "The Industrial City." We add a sentence expressing that fact as an axiom in an applied modal logic; but, we don't want to apply necessitation. We don't want to infer, that is, that Kimberly knows/believes that south San Francisco calls itself "The Industrial City." What does a classy kid like Kimberly care about a place like South San Francisco? We shall have to simply add particular axioms about what Kimberly does (or does not) know/believe about the situation in question; or, better, those facts are part of the situation in question.

The worries about introspection are horses of another color. It is those that I am going to try to honor. One crucial consideration here is sociological. Yuppies simply are not very introspective; they're much too busy networking and consuming to be self-reflecting. The pale cast of introsection surely gets in the way of having good, trendy, expensive fun; one can't get all there is out of driving one's BMW if one is paying attention to one's own thought processes--as opposed to the impression one is making on others of one's kind, etc., etc. Another consideration is a fondness on my part for weak noncommittal systems to which one can add strength--and bold commitments--as one one wishes.

⁹Single subject epistemic/doxastic logics will have two unary modal operators, 'K', 'B', each with a subscript suppressed but both fixed and understood. That is, one is to fix a subject, say Scott, and read 'K' as "Scott knows that..." Of course, if one assumes--as I shall--that all Yuppies are in the relevant respects indistinguishable, one can imagine oneself working with a schematic modal operator, an operator whose subscript is a schematic letter whose substitution instances are singular terms for Yuppies; e.g., names like "Scott," "Kimberly"--not e.g., "Harvey," "Alice."

4. ON KNOWLEDGE.

As noted above, Hintikka argued strenuously for the epistemic version of 4: the thesis that if one knows, one knows that one knows. People attacked this position; Hintikka relented, as well he should have. Most of the bad arguments for skepticism—that is, most of the arguments—have turned on tricking the ingenuous into accepting the thesis that if one knows, one knows that one knows and then arguing that one doesn't know that one knows. Let us suppose that knowledge requires either justification on the knower's part or a "proper" etiology for the belief, e.g. a suitable placement on the knower's part with respect to the fact known (e.g., standing in the right kind of causal relation to it).¹⁰ Surely either of these requirements can be met without the knower's knowing that they're met. Indeed, surely we might sometimes be argued into accepting unreasonably high standards on knowing—so high that though we know, we not only don't know that we know, we actually believe (falsely) that we don't know. Of course, if we're sufficiently gullible, such arguments might even get in the way of the controverted belief (our knowledge of which was in question), so that we cease to know that p because we have (foolishly) ceased to believe it.

For Hintikka's original epistemic logic we can prove that the addition of the axiom schema 4 is equipollent with the addition of the following rule of inference:

RKK: If $I-(Kp \rightarrow q)$, then $I-(Kp \rightarrow Kq)$

For one direction of the proof of equipollence; we have $I-(Kp \rightarrow p)$ (by **T**), whence by **RKK**, we have $I-(Kp \rightarrow Kp)$, whence, by **RKK** yet again, $I-(Kp \rightarrow KKp)$. (The other direction is left as an exercise for the reader.) Imagine that whether Scott knows that p is up for grabs, and let q be any old sentence the truth of which is sufficient for the falsity of the claim that Scott does know that p . Now reason contrapositively and apply **RKK**. To wit;

$(q \rightarrow \neg Kp)$; so $(Kp \rightarrow \neg q)$; so--by **RKK**-- $(Kp \rightarrow K\neg q)$

This may seem innocuous; but it isn't. In order to know that p , poor Scott must know the falsity of anything whose truth rules out his knowing that p . This is precisely the sceptic's trick. Get someone to accept this requirement, and it won't be hard to get that same someone to doubt that anyone knows anything. For the requirement certainly seems to amount to this: if Scott does know that p , then he

¹⁰This supposition encompasses the supposition that knowledge is not just true belief. Much of the recent AI and computer science literature *seems* to suppose that knowledge is just true belief. But it isn't.

knows the falsity of anything whose truth would rule out his knowing that *p*. We might say, then, that Scott, in knowing that *p*, must be in a position to disregard all further evidence with respect to--i.e., in a position to rule out any and all counterpossibilities. But Scott is almost never in a position to disregard all further evidence; so Scott almost never knows anything.

Now all this may be an abuse of the thesis that if one knows, one knows that one knows. (Though I should note that the argument just given is used by Hintikka himself in his--somewhat reluctant--recantation of the axiom. See [Hintikka 70.] Still, I see no reason to accept the thesis. Indeed, I see no reason to accept even the claim that if one knows one believes that one knows. If one does believe that one knows that *p*, one might be said to be certain that *p*. At least, that is how the philosopher G. E. Moore characterized certainty. Provisionally accepting this characterization, I want to say that one can know that *p* without being certain that *p*.

Hintikka also spent time arguing against the epistemic version of **B**:

$$(-K-Kp \rightarrow p)$$

This says that if it is not the case that Kimberly knows that it is not the case that Kimberly knows that *p*, then *p*. This is truly bizarre; a little "introspective ignorance" on Kimberly's part about the scope and limits of her knowledge is going an awful long way. (I suppose her parents--dabbling in epistemic logic--*might* look favorably on this schema; but surely cooler heads would ultimately prevail.) Ruling out **B**, while accepting *K* + **T**, as Hintikka does, provably rules out accepting the epistemic version of **E**:

$$(-K-Kp \rightarrow Kp)$$

That's no great price to pay since the epistemic version of **E** seems wildly too strong. (Thus, by simple transformations, this yields that if one does not know that *p*, then one knows that one does not know that *p*. Would that life were so neat!)

One last word about knowledge and the so-called introspective axioms. I noted in passing that knowledge certainly seems to be more than just true belief. In particular, it seems to require that the belief be justified or that it (and the believer?) stand in some special--perhaps causal--relation to the fact. Eternally controversial issues in the philosophy of knowledge lurk. Let them lurk; it suffices for my purposes to point out that if one buys some version of the second, "causal," account of knowledge--as I am inclined to do--then the knowledge that one knows need not be, in any clear sense, introspective--beyond the bare minimum of knowing that one believes that *p*, if one does. Rather what one must know

to know that one knows that p is that one (or one's mental state of believing that p) stands in the right kind of causal relation to the fact that p . This might involve knowledge about one's sensory apparatus, as well as knowledge about more fully external features of the situation. But this is surely not introspective knowledge at all. (Indeed, there are, I think, similarly external or objective readings of some versions, at least, of the justification story--readings which turn justification-based accounts into "causal" accounts.)

In sum: with respect to the axioms governing the "K" operator, I opt for minimality (modulo some version--restricted or not--of "logical omniscience"). That is, I opt for the epistemic version of $K + T$. The modal core of our epistemic logic is just the modal core of K :

$$\text{II': } K(p \rightarrow q) \rightarrow (Kp \rightarrow Kq)$$

$$\text{R2': } \text{If } \neg p, \text{ then } \neg Kp$$

5. ON BELIEF

As to belief: if no one else and if no one earlier, Freud should have taught us that we don't always know our own minds. Indeed, we can't always believe our minds are as they, sad to say, are. We can believe without believing that we believe; so much for the doxastic version of 4. We can also not believe that we do not believe that p and still not believe that p . That is to say, the doxastic version of **E** seems false:

$$(-B-Bp \rightarrow Bp)$$

Likewise the doxastic version of **B**, which like its epistemic counterpart seems crazed--only more so:

$$(-B-Bp \rightarrow p)$$

If Kimberly doesn't believe that she doesn't believe that p , then p . This is megalomania, even in someone as spoiled as Kimberly is likely to be.

A last word on the standard "introspective" axioms for belief: it can seem as though one's beliefs about one's own beliefs will typically be vouchsafed one by introspection. This seeming gets weaker when one considers past--or future--beliefs of one's own. Certainly for the past, there's memory; but memory of what? Of one's past mental states or of one's past actions? Thus, we often reason as follows: I must have believed that p ; for consider what I did. Independent of Freud, et al., I think there are good reasons for doubting the extent of one's introspective access to one's own current beliefs. Some of these reasons have to do with the nature of the objects of belief; some, with the nature of believing as a state.¹¹ I'm not going to rehearse these here. Instead, I will simply present another scorecard:

AXIOMS RELATING BELIEF AND KNOWLEDGE THAT I ACCEPT

$$Kp \rightarrow Bp$$

AXIOMS RELATING BELIEF AND KNOWLEDGE THAT I DO NOT ACCEPT

$$Bp \rightarrow BBp$$

¹¹I will return to the question of the objects of belief, albeit briefly, below.

$Bp \rightarrow KBp$

$Kp \rightarrow BKp$

$Kp \rightarrow KKp$

6. ON LIMITING INTROSPECTION FOR BELIEF

So, what axioms *do* I want for belief, at least for the beliefs of such as Scott and Kimberly. First, let me remind the reader that, however taken we may be with these Young Upwardly Mobile Professionals, they are not infallible. We cannot allow them the doxastic version of **T** $Bp \rightarrow p$. We might, though, grant them a consistency condition--this comes in especially handy for those whose beliefs are closed under classical tautological consequence. The condition in question is that if Kimberly believes that p , then she does not believe that it is not the case that p . This is a doxastic version of a schema called **D**.

$$(D): \quad (Bp \rightarrow \neg B\neg p)$$

The 'D' is for "deontological" or "deontology." (More Greek.) Deontology is the study of the logic of obligation. The crucial operator there is "it is obligatory that...". It does have a dual: "it is permissible that...". Note that just as we cannot, alas, have a doxastic version of **T** for reasons of fallible belief; so too can we not have a deontological version--for reasons of fallible mores. But we do have it that if it is obligatory that p , then it is permissible that p . That is, if it is obligatory that p , then it is not obligatory that it not be the case that p . This last is just **D**. So **D** is oft regarded as the characteristic deontological axiom. It is, of course, obvious that **D** is a theorem of $K + T$. Is **T** a theorem of $K + D$? We must hope not, for then by granting consistency, we will let in the unacceptable infallibility. How can one tell?

There is one sure way to tell that something is a theorem of a given system--prove it within the system. In general, only infinite patience will avail if one wants, obversely, to show of a sentence that is not a theorem of some system that it is not. Even for decidable systems--and all the logics I will be discussing here are decidable--"direct" proofs of nontheoremhood are really out.

6.1. Model Theory of Intensional Logics

Model theory to the rescue! The model theory of modal logics is good for at least two things: (1) proving in the semantic metatheory that such-and-such is a theorem of so-and-so and (2) proving in the metatheory that such-and-such other thing is not. Logicians, generally, aren't sufficiently silly as to want to work *within* a given formal system; they prefer to work on the outside, using whatever tools are appropriate, to prove things *about* the formal system. This is what Kripke et al. allowed logicians to do with respect to modal logics. The key to Kripke's analysis lies in the introduction of modal models; triples $\langle S, R, v \rangle$ where S is any nonempty set, R is a relation on S , i.e. a subset of $S \times S$, and v is a value assignment meeting standard conditions for standard sentences and the following condition for modal

sentences. Using 'L' now as the strong, necessity-style operator and, the redundant but useful, 'M' as its dual:

L: For any wff. p , and any s in S , $v(Lp, s) = T$
 if $v(p, s') = T$ for every s' in S s.t. sRs' ;
 otherwise $v(Lp, s) = F$.

M: For any wff. p and any s in S , $v(Mp, s) = T$
 if there is at least one s' in S such that sRs' and such that $v(p, s') = T$;
 otherwise $v(Mp, s) = F$.

So the necessity operator is akin to the universal quantifier; its dual, the possibility operator, akin to the existential quantifier. R enters the above as a parameter--as does S , for that matter. What Kripke, et al. showed was that one could ring changes in the nature of R and thereby yield modal models appropriate to different modal logics. One way to think about this is to ignore the value assignments and think of duples: $\langle S, R \rangle$, S and R as before. Call such things *frames*, and go on like this: a formula is valid on a frame just in case it is valid in every model based on that frame--letting v vary. Finally, say that a modal system is **characterized** by a class of frames if all and only its theorems are valid on every frame in that class. *Voila*, different modal systems are characterized by different classes of frames, the difference residing precisely in the conditions on R .¹²

Now, as to why K is called the **minimal normal** modal logic. Simple, K imposes no restrictions on R at all--not even that it be nonempty. So much for minimality. As for normality; here, it's what's **not** in S , as opposed to the nature of R , that counts. Call a subset Q of S **nonnormal** if for every q in Q , every wff. p , and every v , $v(Mp, q) = T$ and $v(Lp, q) = F$. If Q is empty, then S is **normal**. At

¹²Before getting down to some cases, I should bring to the reader's attention my use of the letter 'S', in place of 'W'. The foregoing story is often glossed as follows: let S be a set of possible worlds, and R a relation of relative accessibility between worlds. Necessity is truth in all accessible possible worlds; possibility, truth in at least one. This gloss is just that: the heuristic of thinking of the members of S as possible worlds is, of course, no part of the formal development. Worse, it can be seriously misleading. Don't, dear reader, let it mislead you. In (almost) the immortal words of Brendan Behan:

Don't muck about

Don't muck about

Don't muck about

with

Possible worlds

nonnormal "indices" or "points of evaluation" (each much more appropriately neutral than "possible world"), anything is possible and nothing is necessary. Sounds like fun.¹³

Restricting ourselves to extensions of K , we can speak either of the characteristic condition on R associated with a given schema X or of that associated with the system that consists of $K + X$. I shall speak in the former mode. So here's another scorecard:

SCHEMA	CONDITION ON R
T	$(s)(s R s)$ <i>reflexivity</i>
4	$(s, t, u)(s R t \ \& \ t R u \ \rightarrow s R u)$ <i>transitivity</i>
B	$(s, t)(s R t \ \rightarrow t R s)$ <i>symmetry</i>
E	$(s, t, u)(s R t \ \& \ s R u \ \rightarrow t R u)$ <i>Euclidean condition</i>
D	$(s)(\text{Et})(s R t)$ <i>seriality</i>

It is now obvious that if R is reflexive it is serial and just as obvious that R can be serial without being reflexive. So, $K + \mathbf{T}$ yields \mathbf{D} ; but $K + \mathbf{D}$ does not yield \mathbf{T} . We're safe; Scott and Kimberly can be logically omniscient and consistent, at least with respect to their beliefs, without being infallible.

6.2. Some Applications.

Now that we have some tools at our disposal, there are other conditions we might want to consider. Scott and Kimberly, after all, think mighty highly of themselves; perhaps, although they are not infallible, they think they are. One expression of this unseemly immodesty--nay, arrogance--is \mathbf{U} :

$$\mathbf{U}: \quad (\mathbf{B}(\mathbf{B}p \rightarrow p))$$

We shall reject \mathbf{U} . To give it its model theoretic due; there corresponds the following nameless characteristic condition on R :

¹³Nonnormal worlds--frames containing such--enter into the semantics of Lewis's $S1 - S8$ --the differences among these being correlated with differences in the accessibility relationship. No one has ever taken these systems very seriously as logics of necessity and possibility. Of course, if I may remind the reader of the fabulous nature of the fable with which I began, they were not meant to be such. I should also note that frames with "impossible" possible worlds have been looked to for a way of handling, within modal logic, the problems of logical omniscience. I will have nothing to say about such attempts in *this* essay. See [Hintikka 75].

$$(s, t)(s R t \leftrightarrow t R s)$$

One can show that **D** and **U** are independent. To show that **D** does not yield **U**, consider the following frame:

$$S = \{s, t\} \quad R = \{\langle s, t \rangle, \langle t, s \rangle\}$$

Here R is *serial*, but not **U**-ish. (Does it *look* **U**-ish?) To get a frame which satisfies **U** but not **D** is but a moment's work:

$$S = \{s, t\} \quad R = \{\langle s, s \rangle\}$$

Note that this frame is not *reflexive*, that is, not **T**-ish. So, **D** and **U** are independent and both are weaker than **T**. Indeed, $K + \mathbf{D} + \mathbf{U}$ is a system strictly weaker than $K + \mathbf{T}$, for note the following:

$$S = \{s, t\} \quad R = \{\langle s, t \rangle, \langle t, t \rangle\}$$

6.3. A Few Doxastic Paradoxes

Before leaving **U** behind, I should note the connection between it and the so-called "Paradox of the Preface." **U**, as noted, is too much; not even Scott and Kimberly believe they are infallible. So, both Scott and Kimberly believe that one of their beliefs is false. But then not all of their beliefs could be true.

Take Scott. He's a reasonable fellow, as Yuppies go. He believes that at least one of his beliefs is false. That is, not only does he not conform to the self-regarding standards of **U**; he positively repudiates same. Now either this belief in his own fallibility--call it **non-U**--is false or not. If it is false, then at least one of his beliefs is false--viz. **non-U**; so **non-U** is true. And, of course, if **non-U** is true, then at least one of his (other) beliefs is false. So whether **non-U** is true or false; it is true. So Scott's belief that at least one his beliefs is false must be true; so at least one of his beliefs is false. **non-U** is fated to be true.

Let's go more slowly here. Let's assume that the "range" of **non-U** does not include **non-U** itself. That is, Scott believes that at least one of his beliefs other than **non-U** is false. Suppose, for simplicity's sake, that Scott has finitely many other such beliefs: p, q, r, \dots Suppose, further, that Scott's beliefs are closed under (finite) adjunction. (This, by my lights, is likely to be a wild supposition; in general, the supposition of unrestricted adjunction--for any attitude--is an extremely dubious one. This is one reason for being dubious about K .) Scott, then, believes the conjunction of p with q with r, \dots But he also believes **non-U**; this is to believe: either not- p or not- q or not- r or... But these two beliefs are inconsistent. Neither

one of them is **non-U**: at least one must be false; so **non-U** is true. Of course, although the second of the two beliefs--the disjunction of the negations of the conjuncts of the first--is not itself **non-U**, it records--in the context of the finitely many other beliefs conjoined in the first--the effect of believing **non-U**.¹⁴

The situation is even more baroque if we put **non-U** back into the pot of Scott's beliefs. Indeed, it is the situation as outlined two paragraphs ago. The supposition that **non-U** is not true leads immediately to the conclusion that it is true. But we needn't stop there. Return to the troublesome case where all of Scott's other beliefs are true. If **non-U** cannot but be true, then it is true. But then all Scott's beliefs are true, after all. But then **non-U** is false, after all. Something is wrong somewhere.

What seems to be wrong is that Scott, no matter how hard he tries, can't successfully believe--either truly or falsely--that at least one of his beliefs is false *unless* one of his other beliefs is false. In which case, of course, no matter how hard he tries, Scott can't help but believe truly that at least one of his beliefs is false--if he believes it at all. Again if Scott were ever successfully to believe the *first-person* version of the negation of **non-U**, then *that* belief would be guaranteed to be true. The first-person version of **U** is a bit much--even for Scott; the third person version, a bit much even for his parents. The third person-version of **non-U** seems just fine--surely it is not the case that Scott believes that if Scott believes that *p*, then *p*. Finally, the first-person version just cannot be falsely believed.¹⁵

So much for **U**. There is another paradox about: to wit, Moore's paradox. (Arguably the first pragmatic paradox to be remarked upon.) Let's pick on Kimberly this time. Kimberly, poor lass, has false beliefs. So we will have occasion to say such things as:

*Kimberly believes that *p*; but it is not the case that *p*.*

Moreover, Kimberly is not omnidoxastic; there are truths she simply does not believe. (I will speak of the trait of believing all the truths there are as *omnidoxasticity*.) So we will have occasion to say such things as:

**p*; but Kimberly doesn't believe that *p*.*

Kimberly, moreover, believes that she has false beliefs--if you don't believe me, advert to the above and ask Scott. But, notice how odd it would be for her to say:

¹⁴I shudder with this talk of conjoining and disjoining beliefs; still, it's a convenient shorthand -- but *for what*?

¹⁵This discussion of the Paradox of the Preface is just a retelling of a tale told, in Polish notation, by A.N. Prior. [Prior 71.] The Paradox was first noticed by D.C. Makinson.

I believe that p; but it is not the case the p.

It is perhaps odder even for her to come out with the first-person denial of omnidoxasticity:

p; but I don't believe that p.

G. E. Moore first pointed out the paradoxical character of the first-person versions of what, in third-person forms, are completely innocuous things to say. I have said that the schematic letters that come with of our 'B' and 'K' operators were to have singular terms for subjects as substitution instances. "I" is such a singular term, but a very special one. Note that even if your name were Kimberly--and you were alone in being so named--it could be perfectly nonparadoxical for you to say:

p; but Kimberly doesn't believe that p.

You might, after all, not know your own name, might not--in this sense--know who you are. Without going much more deeply into problems about indexicals and quasi-indexicals, we cannot really go very deeply into Moore's Paradox; so, in what follows, I am going to be playing a little fast and loose. I am going to assume that Scott knows who he is, at least in so far as he knows that he is (the one and only) Scott--the one and only person named 'Scott', *mutatis mutandis* for Kimberly. In this respect, I follow Hintikka's lead.

6.4. Moore's Paradox and the Schema Y

To return to the main line, the key here is the following schema:

(p & -Bp)

We cannot rule it out by ruling in its denial:

-(p & -Bp)

for that is equivalent to the wholly unacceptable **O**:

(p --> Bp) *omnidoxasticity*

Note that the negation of our version of "*I believe that p; but it is not the case that p*", is the equally unacceptable infallibility axiom **T**: $(Bp \rightarrow p)$.

What we want to rule in is

$$(\neg B(p \ \& \ \neg Bp))$$

It is not the case that Scott believes both that *p* and that it's not the case that he (Scott) believes that *p*. This is equivalent to:

$$(\neg B(p \rightarrow Bp))$$

It is not the case that Scott believes that it is not the case that if *p*, then Scott believes that *p*. That is, though Scott does not believe in his own omnidoxasticity, it is not the case that he believes that it is not the case that he is omnidoxastic. Another perspective on this dark saying is vouchsafed us by distributing "B" over "&" in the earlier version:

$$(O'): \quad (\neg(Bp \ \& \ B\neg Bp))$$

It is not both the case that Scott believes that *p* and that he believes that it is not the case that he believes that *p*.

This last raises the question of **U** again. I have simply assumed that Scott does not believe he is infallible. So we do not accept

$$(U): \quad (B(Bp \rightarrow p))$$

But we can deny that Scott believes the negation of the infallibility schema. We can allow

$$(\neg B(Bp \rightarrow p))$$

This is equivalent to

$$(-B(Bp \ \& \ -p))$$

It is not the case that Scott believes both that he believes p and that it is not the case that p. Or distributing "B" over "&":

$$(U'): \quad (-(BBp \ \& \ -Bp))$$

It is not the case both that Scott believes that he believes p and it is not the case that he believes p.

Perhaps the basic drift is now clear. I do not want to buy the standard "axioms of introspection"; **not even for belief**. Rather, the logic of belief I am proposing is generated by the intuition that what one wants is that one's subjects--Scott, Kimberly--not be stuck with certain kinds of false introspective beliefs. So, I propose that they not make certain kinds of mistaken self-ascriptions of belief; thus, that they not both believe that p and believe that they do not believe that p. Again, they should not both not believe that p and believe that they believe that p. To grant this freedom from error is already a generous gesture of idealization on my part; but, of course anyone as blithely unconcerned with "logical omniscience" as I cannot blanch at idealizing. Still, it is a much weaker form of idealization than guaranteeing oodles of true self-ascriptive beliefs. Yuppies, remember, don't introspect much.¹⁶

The key idea in the above might be put as follows: take a controversial schema and deny that Scott or Kimberly believes its negation. This is exactly how we got **O'** from the omnidoxastic schema **O**; and **U'** from the obnoxiously self-satisfied **U**. Let's apply this algorithm to the doxastic versions of both **4** and its converse, the unnamed

$$(BBp \ \rightarrow \ Bp)$$

Let's name this **C4**, for the converse of **4**. This is not to be confused with **U**. (Though it is entailed by, it does not entail, **U**.) What we get, after the standard transmogrifications, are **4'** and **C4'**

$$(4'): \quad (-B(Bp \ \& \ -BBp))$$

¹⁶ Of course, some true self-ascriptions creep in with the logic, with *K* itself. For instance, by **R2**: $I-(p \rightarrow p)$, so $I-B(Ip \rightarrow p)$; so $I-B(BIp \rightarrow p)$. Voila, introspection! But nothing to write home about.

$$(C4'): \quad (-B(BBp \ \& \ -Bp))$$

And, if one thought it more perspicuous:

$$(4'): \quad -(BBp \ \& \ B-BBp)$$

$$(C4'): \quad -(BBBp \ \& \ B-Bp)$$

Yuppies may not be introspective; but they are **confident**--even about the rare introspective beliefs they may entertain. Although it is not the case that if one believes that p, then one believes that one believes that p, neither is it the case that one believes both that one believes that one believes that p and yet does not really believe that p. (That was 4', in case you couldn't guess.) Moreover, it isn't even true that if one believes that one believes that p, then one does believe that p. But it is true that one doesn't believe both that one believes that one believes that p and yet that one doesn't believe that p. (That's C4'.)

Let's look back at O', the one prize we captured from our perusal of Moore's Paradox:

$$(O'): \quad -(Bp \ \& \ B-Bp)$$

This is equivalent to

$$(-B(p \ \& \ -Bp))$$

which is, in turn, equivalent to

$$(Y): \quad (Bp \ \rightarrow \ -B-Bp)$$

If Kimberly believes that p, then it is not the case that she believes that it is not the case that she believes that p; more colloquially: if she believes that p, then she doesn't believe that she doesn't believe that p. *Nota bene*: no **real** introspection is required; rather, what is being ruled out is that Kimberly have certain kinds of false introspective beliefs.¹⁷

¹⁷The contrapositive of Y is $(B-Bp \rightarrow -Bp)$. If Kimberly has any positive introspective belief to the effect that she does not believe that p, then she does not believe that p. Note the asymmetry here between negative and positive introspective beliefs. As Achilles said to the Tortoise, "That's Classical Logic".

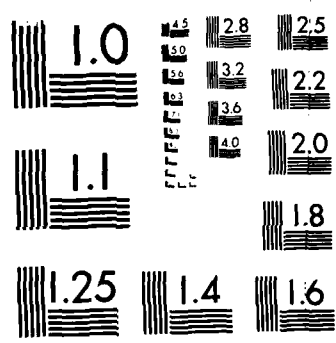
KNOWLEDGE REPRESENTATION AND NATURAL-LANGUAGE SEMANTICS
(U) SRI INTERNATIONAL MENLO PARK CA R C MOORE AUG 85
AFOSR-TR-85-1098 F49620-82-K-0031

R C MOORE AUG 85

F/G 5/7

NL

FN



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS 1963 A

If we add this schema, which we have dubbed **Y** for the obvious reason, we have a modal system that will yield all of **D**, **U'**, **4'**, **C4'**, and none of **O**, **U**, **4**, or **C4**.¹⁸

6.5. More on **Y**

Y is, of course, the doxastic version of the nameless (**Lp** \rightarrow **MLp**). This latter is just an instance of (**p** \rightarrow **Mp**), which is a fairly basic principle about possibility. Indeed it is just the other side of the coin from **T**. But we don't have **T** for belief; nor do we have (**p** \rightarrow **-B-p**). We have **Y**.

No doubt the reader is just dying to get a gander at the characteristic condition on **R** associated with **Y**. Take a gander:

$$(Y): \quad (s)(Et)(s R t \ \& \ (u)(t R u \rightarrow s R u))$$

That any frame which is **Y**-ish is *eo ipso* **D**-ish—that is, *serial*, is obvious. The converse does not hold. Consider:

$$S = \{s, t, u\} \quad R = \{ \langle s, t \rangle, \langle t, u \rangle, \langle u, s \rangle \}$$

This is serial but not **Y**-ish. Thus, $s R t$ and $t R u$; but it is not the case that $s R u$. Moreover, **Y** does not yield **U**. (Remember, we don't want it to.), thus:

$$S = \{s, t, u\} \quad R = \{ \langle s, t \rangle, \langle t, u \rangle, \langle s, u \rangle, \langle u, u \rangle \}$$

Note that though $s R t$, it is not the case that $t R t$. Indeed this shows that **Y** does not yield the unwanted **T**.

¹⁸NOTA BENE: Craig Harrison, in a discussion of the paradox of the unanticipated examination, has argued for a modal logic of belief much like the one proposed here. See [Harrison 80]. Actually, his proffered alternative is weaker; it is essentially **K** + **D**. But he, too, considers the schema I have called **Y**. Moreover, he, too, adduces Moore's Paradox as, at the very least, a consideration. The history here is complicated. The work on what is now *Yuppie logic* began almost fifteen years ago, after Harrison's paper appeared. When I began the work, I hadn't yet read Harrison's paper. Indeed, I wasn't thinking about the paradox of the surprise exam at all. Then, as in the present essay, I ignored all issues of time and its passage; then, as in the present essay, the considerations for and against various principles had their source in Moore's Paradox, (to a lesser extent) the Paradox of the Preface, and general epistemological considerations. In fact, I think Harrison's treatment of the unanticipated or surprise exam extremely interesting, but—in the end—inadequate. He bases too much on a rejection of the theses that if one knows/believes, one knows/believes that one does. I, too, reject those, though not simply (or at all) because that decision allows one to hold that the set up in the surprise exam puzzle is a consistent one. Though Harrison treats of time indexed epistemic and doxastic operators, he doesn't do enough with them, doesn't say enough about the principles that should govern them. In any event, I hope to address the issues raised by that paradox in the future. Still, I don't want it thought that the idea of looking at intensional logics for belief and knowledge which extend **K** but not as far as any of the standard logics of necessity do, is either unique with, or original to, me. Harrison, and no doubt others, including Binkley [Binkley 68], got there first.

As to $(BBp \rightarrow Bp)$: Its characteristic condition is as follows:

$$(BBp \rightarrow Bp) \quad (s,t)(s R t \rightarrow s R^2 t)$$

So consider the frame:

$$S = \{s, t, u\} \quad R = \{\langle s, t \rangle, \langle t, u \rangle, \langle s, u \rangle, \langle u, u \rangle\}$$

Here, $s R t$; but it is not the case that $s R^2 t$. That is, there does not exist an s' , s.t. $s R s' \& s' R t$.

Finally, as to 4. $(Bp \rightarrow BBp)$:

$$S = \{s, t, u\} \quad R = \{\langle s, s \rangle, \langle t, t \rangle, \langle u, u \rangle, \langle s, t \rangle, \langle t, u \rangle\}$$

$s R t$ and $t R u$; but it is not the case that $s R u$. (This particular frame is reflexive; but, of course, not all Y-ish frames need be.)

I assume, by the way, that it's obvious that Y yields neither B nor E nor O. The characteristic condition of this last is: $(s,t)(s R t \rightarrow s = t)$.

So much for the crucial negative results. Let's think positively. We've already noted that Y--that is, $K + Y$ --yields D. Y yields O', because it is O'. There are fairly straightforward direct proofs in $K + Y$ of U', C4' and 4'.¹⁹

The reader may well wonder about the results of applying the above treatment to B and E. That is, what about the schemata that result by negating the believability--for such as Scott and Kimberly--of their negations? The resulting schemata are, in order, B':

$$(-B(-B-Bp \& -p))$$

or:

$$(-(B-B-Bp \& B-p))$$

¹⁹ Rather than bore the reader to tears with such proofs, I'll give hints for their construction. To prove U', simply substitute 'Bp' for 'p' in Y; to get 4' from U' is the work of but a moment, making use of the same substitution pattern as before--'Bp' for 'p'. Finally, to get C4': use axiom scheme II of K on D, put the result of that together with Y, and *Voilà*, C4'.

and E' :

$$(-B(-B-Bp \ \& \ -Bp))$$

or:

$$(-(B-B-Bp \ \& \ B-Bp))$$

These are sufficiently opaque as to not be worth much worry; but, in fact, they are both theorem schemata of $K + Y$.²⁰

²⁰Proofs left as nontrivial exercises for the reader.

7. SUMMING UP

It is time both to sum up and attempt, at least, to see YUPPIELOGIC from a proper perspective. Any "logic" of knowledge and belief will have to be based on idealizations. There are, at least, two orthogonal dimensions along which to idealize. One dimension is that of the logical competence of the subject knowers/believers. The other is that of the degree to which the subjects have knowledge of or beliefs about their own knowledge and beliefs. In this essay, I have decided to idealize quite recklessly along the first dimension. I have, of course, allowed idealization along the second as well, but much less than the norm. The guiding intuition all along has been that, with respect to their attributions to themselves of knowledge and especially belief, the axiomatization should guarantee our subjects against certain kinds of epistemic/doxastic grief--*not that it should guarantee them all manner of epistemic/doxastic success*. Imagine a subject whose beliefs conform to our account. Such a subject will be under no pressure to change her beliefs about her beliefs--*no pressure, that is, stemming from conflicts between what she believes about what she believes and what she actually believes*. I assume, of course, that *falling short of "introspective omniscience" by itself generates no pressure, and no such conflicts*.

Let's return once again to O and $O' (= Y)$. O is a completely general schema to the effect, roughly, that our subject--Kimberly, say-- believes every true proposition. This is obviously bonkers. We allowed, however, that Kimberly does not believe the negation of O . This yielded O' , and O' simply denies that Kimberly believes things and also believes that she doesn't believe those things. It denies that Kimberly is subject to a certain kind of error of self-attribution--one might say *the basic kind* of such error. Note that by necessitation, Kimberly will of course believe that she is not thus subject to that kind of error. That is, she will believe, not that she has any real talent for doxastic self-attribution or introspection, but that she doesn't go around believing that she doesn't believe things she actually does believe.

Another way to see what's going on is to go farther than I have so far in intermixing belief and knowledge. At the moment the only two-operator schema I allow is to the innocuous effect that knowledge requires belief. In passing I mentioned Hintikka's argument against the epistemic version of B . B is sufficiently bizarre that one should not require even Scott to believe it; but what if we try out our trick on it? What about:

$$(-B(-K-Kp \rightarrow p)) \quad ?$$

What indeed? Let's transmogrify, using our recipe:

$$(-B(-K-Kp \ \& \ -p))$$

It is not the case that Scott believes both not- p and that he does not know that he doesn't know that p . If he knew that p , he would not believe that not- p . (By **D** and the requirement that knowledge involves belief.) Of course, if he knew that he didn't know that p , he might very well believe that not- p . (Or not: he might be open-minded, have no opinion, with respect to the question.) If he doesn't know that he doesn't know that p , he might still believe that not- p . After all, he just might not know that p , for instance, because he doesn't believe that p , but not know that he doesn't know it—for instance because he doesn't believe that he doesn't believe it. But if he were to believe that he doesn't know that he doesn't know that p --say, because he doesn't know that he doesn't believe that p --and yet still believe that not- p , then he would have reason for concern lest he be inconsistent, or of two minds about his attitude toward p (or its negation). And we have ruled out such worries.

Try another transform:

$$(-(B-K-Kp \ \& \ B-p))$$

Either Scott doesn't believe not- p or he doesn't believe that "for all he knows", he knows that p --where, a la Hintikka, I'm reading ' $-K-q$ ' as "for all Scott knows, q ". So, imagine Scott believes that not- p . Then he had best not believe that for all he knows, he knows that p .

This trick works for the epistemic versions of **4** and **E** as well. No doubt looking at one of these will suffice. Let's do **4**, which--in its epistemic version, of course--was the most hotly contested of the "introspective axioms" originally proposed by Hintikka.

$$(-B(Kp \ \& \ -KKp))$$

Scotty should not believe both he knows that p and that he doesn't know that he knows it. It is quite possible that Scotty know that p without knowing that he knows it. Remember, we reject **4**. But if he should believe that he knows that p , then it will not do to believe that he doesn't know that he knows it. Identifying Scott's being certain that p with his believing that he knows that p : if Scott is certain that p , then he doesn't believe that he doesn't know that he knows that p . (Although, again, he really might not know that he knows it.) He would not continue to be certain that p if he believed that he didn't know that he knew that p . Put otherwise: being certain that p requires not believing that for all you know you might not know that p .

8. SOME FINAL SCEPTICAL REMARKS.

Now to say a word about believing and knowing--in particular about believing. Believings and beliefs come in a wide variety of "modes". One talks of explicit and implicit beliefs, of conscious and unconscious beliefs, of occurrent ("active") and dispositional beliefs. These three dimensions/dichotomies are very likely independent, and there may be other such dimensions or dichotomies. To which of these, if any, is our 'B' operator supposed to correspond? Hintikka, for instance, clearly intends his 'B' and 'K' operators for what he calls "active belief" and "active knowledge." But he also seems to suppose that being active involves being conscious; that is, being an active belief involves being a belief of which the believer is conscious. Further, he argues--naturally enough--that it is only a certain mode (or modes) of believing for which various of his principles and rules are appropriate. Thus, for instance, the doxastic version of 4, that if one believes that p, one believes that one believes that p, holds of active, conscious beliefs. (He thus rules out of court--he thinks--references to Freud, self-deception, and the like.)

8.1. *On Belief States*

I can be no more than brief here, but it seems to me that a much more important "dichotomy" is that between two different conceptualizations of the role of belief. According to one conceptualization, the main locus or arena of beliefs is in thinking that is aimed at truth, that is, in "theoretical reasoning," considered in abstraction from the creature's possibilities of and requirements for action. This conceptualization leads quite naturally to focussing on conscious beliefs, consciously arrived at, and thereby to focussing on language using creatures who can express their beliefs, including their beliefs about their own mental states. The other conceptualization might be called "functional"; Robert Stalnaker has called it "pragmatic-causal"[Stalnaker 85]. Here the main arena is action; the fundamental role of beliefs in the mental life of believers is as states that, together with desires and intentions, guide or direct or determine behavior. Roughly, to say that a subject believes that p is to say that if the subject were to desire that q, then he would be disposed to act in a way that would bring it about that q were it to be the case that p. This conceptualization of beliefs is essentially dispositional; within it, being active means playing the characteristic role of belief in an actual behavioral episode and has nothing whatsoever to do with being conscious--let alone with being linguistically expressible. Again, within this conceptualization, neither language nor language users occupy any special privileged position of interest.

I take it that it is clear enough that a concern with "introspection" goes most naturally with the first of these two ways of thinking about belief. This is true even if one clearly distinguishes "introspective beliefs" from a subject's beliefs about its own mental states. Let me now clearly distinguish these two. The second has solely to do with the content of beliefs; a rough and ready characterization is simply this: the subject-matter of the creature's belief is about that creature's own mental states--including its own *present* mental states. Even here, and even in the case of beliefs about one's own

present mental-states, one can distinguish beliefs about one's own mental states in the "first-person mode" and in the "third-person" mode. So, for example, I might believe that the sixth oldest researcher in AI believes that *p*, without realizing that I *am* the sixth oldest researcher in AI. If so, I might be said to have a third-person belief about my own beliefs. In general, one can concoct examples in which a creature can have beliefs about itself without realizing that it is the very thing at which the beliefs in question are directed.

"Introspective beliefs" on the other hand are beliefs about one's own mental states that are caused in a certain way (or ways), or which arise out of the functioning of one (or another) specific--though perhaps completely unspecified--cognitive mechanism called "introspection."²¹ If one is thinking about beliefs from the "pragmatic-causal" perspective, it's hard to get excited about "introspective beliefs" unless one simply assumes that all beliefs that arise out of introspection are "in the first-person mode". But, then, what's crucial about them is that latter fact not their etiology.

Indeed, from within the "causal-pragmatic" or "functionalist" tradition, it's hard to get excited about epistemic/doxastic logic.²²

8.2. *On the Contents of Beliefs*

Let me now say a word about the objects or contents of believings: that is, about beliefs. If the objects or contents of such mental states as believing and knowing are to be truth-valuable--as they are represented as being in all standard epistemic and doxastic logics, then they had best make or correspond to or just *be* determinate claims upon reality. Sentences--sentences types--of natural languages precisely **do not** correspond to or make such claims. Sentences--better, well formed formulae--of standard logical languages, by tacit conventions of interpretation or of intended range of applicability, are supposed to make such determinate claims. That is, such sentences are supposed to be eternal: any statement-making utterance of such a sentence yields the same propositional upshot, makes the same determinate claim upon the world. (Of course, the worlds in question are typically conceived of as mathematical structures, that is, as consisting of eternal or timeless objects standing in certain timeless relations among themselves.) So if we imagine a subject whose beliefs are mediated (carried) by, as well as being expressible in, sentences of such a formal language, that is, if we imagine the subject's believing that *p* as

²¹Of course, given this characterization, it is really an open question whether there are any introspective beliefs. I think that there are, but that the members of only very few species can have them. I used to think that only the members of language-using species could, I am now prepared to be more liberal and include those species which manifest a certain kind and degree of social or group organization. Unfortunately, I can't characterize this kind or that degree; nor do I have any good argument as to the necessity of the alluded to condition. But then again no one has ever told me of what introspection really consists.

²²NOTA BENE: from within the "pragmatic-causal" world picture, what's crucial seems to be the "first-person" mode of self-attribution in that mode which guides action. Or, perhaps one should say that what is crucial is the relation between the first-person and the third-person modes of self-attribution. See, e.g., [Perry 85].

involving the subject's saying--to himself, for example--a sentence of such a language, then we *might* have no trouble convincing ourselves that the content of such a subject's beliefs are transparent, completely accessible, to that subject. This is again precisely what we cannot imagine even if we follow this language-involving conception of belief but think instead of our subject thinking to itself in (using) sentences of some natural language. (In all of this, I am assuming complete semantic competence--though, of course, without having a complete theory of what constitutes such competence. At any rate, I am assuming--for the sake of argument--that such competence consists, at least, in the subject's knowing what any sentence of his language "means"; so, in the case of a language with only eternal sentences, in knowing for every sentence, what claim is made by that sentence--what the world would have to be like for that sentence to be true.) If we deny that believing is essentially language involving, it is harder still to see why or even how the content of a subject's beliefs should be transparent to that subject.

Moreover, if we are working within the functionalist paradigm, we will see that--in so far as we are interested in generalizations across subjects or across time and changing circumstances--our primary interest will be in a notion of content under which contents are not truth-valuable and do not correspond to determinate claims upon reality. Note, well, that I speak of "content", not of "object: I don't think there is a useful sense in which the meanings of non-eternal sentences are objects of belief. We shall, that is, be interested in a notion of content such that (e.g.) when both Scott and Kimberly say to themselves, "There's no milk in the fridge," even if at different times and locations, and the like, the mental states that such imagined sayings indicate have the same content. For if both desire to drink some milk, or even if both desire that there be some milk in the fridge, then they would be disposed to act in such a way as to bring it about that there would be milk in the fridge were it the case that there was as yet no milk in the fridge. Just as the truth-valuable contents of their mental states are different, so too are the contents (objects) of their desires, both their desires *to* (drink some milk) and their desires *that* (there be milk in one's fridge). Note, too, the talk of "act in such a way." The way or ways in question can only be characterized at a level of abstraction or generality that cuts across the differences in the actual circumstances of Scott and Kimberly and cuts across them in a way correlative to that in which the sameness of their mental states cuts across the differences in the truth-valuable contents/objects of their beliefs. Much mischief has been wrought by failure to distinguish these two different conceptions of content [Barwise and Perry 83]. Finally and to repeat: from within the functionalist perspective, it is the second notion of content that is crucial, or--again--the relation between the two notions. Hence again, what interest could there be, from within such a conceptualization, in standard epistemic/doxastic logics--formalisms which, at least standardly, take it that the proper objects of belief, within the logic, are truth-valuable and propositional?²³

²³Here I should remind the reader that within epistemic and doxastic logics, belief and knowledge don't really get treated as relations to propositions; that is, such logics are to be contrasted with theories--say, first or higher order theories--of the relations in question. In the context of these intensional logics, the relations are metatheoretic epiphenomena, arising out of a particular heuristic for understanding a particular model-theoretic treatment.

It may be, then, that to take epistemic/doxastic logics seriously, one must both be working from within that conceptualization of cognitive states according to which they are either essentially or importantly language involving and, further, conceive of the language(s) in question on the model of standard formal languages, as consisting, that is, of eternal sentences only. This *could* be taken as an argument to the effect that the proper home of epistemic/doxastic logic is theoretical computer science--precisely the locus of its greatest current vitality.

References

- [1] Anderson, A. and Belnap, N.
Entailment, Volume I
Princeton University Press, Princeton, NJ, 1975
- [2] Barwise, J. and Perry, J.
Situations and Attitudes
Bradford Books, MIT Press, Cambridge, MA, 1983
- [3] Binkley, R.
The Surprise Examination in Modal Logic
Journal of Philosophy 65 (1968) pp. 127-136
- [4] Fagin, R. and Halpern, J.
Belief, Awareness, and Limited Reasoning
In **Proceedings of the Ninth International Joint Conference on Artificial Intelligence**
1985
- [5] Fagin, R. and Vardi, M.
An Internal Semantics for Modal Logic
In **Proceedings of the Seventeenth ACM Conference on Theory of Computing**
1985
- [6] Halpern J. and Moses, Y.
Knowledge and Common Knowledge in a Distributed Environment
In **Proceedings of the Third ACM Conference on Principles of Distributed Computing**
1984
- [7] Harrison, C.
The Unanticipated Examination in view of Kripke's Semantics for Modal Logic
In **Philosophical Logic**, ed. J. Davis, D. Hockney, and W. 8}Wilson, D. reidel, Dordrecht, 1969
- [8] Hintikka, J.
Knowledge and Belief: An Introduction to the Logic of the Two Notions
Cornell University Press, Ithaca, NY, 1962
- [9] Hintikka, J.
Impossible Possible Worlds Vindicated
J. of Philosophical Logic 4 (1975), pp. 475-484
- [10] Hintikka, J.
"Knowing that one knows" Reviewed
Synthese 21 (1970)
- [11] Lemmon, E. J., in collaboration with D. Scott
The "Lemmon Notes": An Introduction to Modal Logic

ed. K. Segerberg
Blackwell, Oxford, 1977

- [12] Lewis, C. I.
A Survey of Symbolic Logic
University of California Press, Berkeley, CA, 1918
(reprinted, with excisions, by Dover, 1961)
- [13] Perry, J.
Perception, Action, and the Structure of Believing
to appear in a **Festschrift for Paul Grice**, ed. R. Grandy
and R. Warner, Oxford University Press, Oxford, 1985
- [14] Prior, A. N.
A Budget of Paradoxes
in **Objects of Thought**, ed. P. T. Geach and A. J. P. Kenny,
Oxford University Press, Oxford, 1971
- [15] Stalnaker, R.
Inquiry
Bradford Books, MIT Press, Cambridge, MA, 1984

Appendix G

PRELIMINARY REPORT ON A THEORY OF PLAN SYNTHESIS

SRI International



PRELIMINARY REPORT ON A THEORY OF PLAN SYNTHESIS

Technical Note 358

August 1985

By: Edwin P.D. Pednault
Artificial Intelligence Center

**APPROVED FOR PUBLIC RELEASE:
DISTRIBUTION UNLIMITED**

The research reported herein was supported by the Air Force Office of Scientific Research under Contract No. F49620-82-K-0031, by the Office of Naval Research under Contract Nos. N00014-80-C-0296 and N00014-85-C-0251, and through scholarships from the Natural Sciences and Engineering Research Council Canada and le Fonds F.C.A.C. pour l'aide et le soutien à la recherche, Quebec, Canada.

333 Ravenswood Ave. • Menlo Park, CA 94025
415 326-6200 • TWX 910-373-2046 • Telex 334-486

Introduction

Classical planning problems have the following form: given a set of goals, a set of actions, and a description of the initial state of the world, find a sequence of actions that will transform the world from any state satisfying the initial-state description to one that satisfies the goal description. In principle, a problem of this type may be solved by a very simple procedure: merely enumerate all possible sequences of actions and test each until one is found that achieves the intended goals. By this procedure, we will eventually find a solution if one exists. However, in practice, not only do we want to find a solution, we want to do so expeditiously. Quick and efficient problem solving is desirable primarily for reasons of economy: the less time it takes to solve a problem, the more productive one can be. Furthermore, in some situations, the time it takes can mean the difference between success and failure, as is the case when the problem is part of a scholastic exam or when the problem is to prevent meltdown in a nuclear reactor.

Previous work aimed at developing efficient planning techniques has been highly experimental in nature, the standard methodology being to explore ideas by constructing computer programs. For the most part,¹ very little theoretical analysis has been done to determine why the programs work, when they are applicable, and whether they can be generalized to solve larger classes of problems.

In my thesis [8], I venture to the opposite extreme and examine the question of efficient planning from a rigorous, mathematical standpoint. My analysis is based on the premise that one of the main impediments to efficient planning is search, and that exhaustive search can be avoided only if

¹ The exceptions to this are Warren's analysis of his WARPLAN program [17] and, just recently, Chapman's logical reconstruction of nonlinear planning [2, 3]. Warren's analysis is primarily concerned with proving the correctness of WARPLAN. Chapman, on the other hand, has analyzed previous work in nonlinear planning and, on the basis of this analysis, has constructed a program called TWEAK that is provably correct.

the problem being solved has properties that can be exploited to constrain the search. Accordingly, my methodology has been to construct a mathematical framework in which to study planning problems, to explore this framework for theorems that can be used to constrain the search for a solution, and then to construct planning techniques based on the theorems found. The techniques are described in precise, mathematical terms and are capable of solving any problem that may be expressed in the framework, provided a solution exists. While the techniques may be implemented in a straightforward manner, there are a number of implementational issues identified, but not addressed, in my thesis that need to be resolved before an efficient program can be obtained.

Although we have been working independently and in parallel, my work can be viewed as a significant extension of work recently reported by Chapman [2, 3]. While our approaches are similar, the framework I have developed encompasses a much broader class of problems and addresses some of the representational issues that Chapman identifies. In addition, I have been able to unify many more ideas in automatic planning and show how they arise from first principles. These ideas include not only nonlinear planning [11, 12, 15, 19], means-ends analysis [4], and opportunistic planning [6], which are incorporated into Chapman's technique, but also goal protection [14, 16, 17], goal regression [9, 16], constraint formulation and propagation [12], and hierarchical planning [10, 11, 12, 15, 19].

This report is intended to provide a glimpse of my thesis research. Only about a quarter of the topics presented in my thesis, however, are covered here. It would therefore appear advisable at this point to summarize the topics I have included and those I have not.

In the next chapter, an intuitive explanation of the mathematical framework is provided and a language introduced for describing the effects of an action. In the framework presented here, actions are assumed to be deterministic—in the sense that performing an action transforms the world from its current state to a uniquely determined succedent state. The synthesis techniques, however, do not require determinism, and in my thesis I present a more general framework that permits actions to be nondeterministic.

The language for describing actions is interesting in that it combines the generality of the situation calculus [7] with the notational convenience of STRIPS [5]. This allows the frame problem of the situation calculus to be circumvented to the same extent that it can be done in STRIPS. As I show in my thesis, but not in this report, any problem that can be described in the situation calculus has an equivalent formulation using this language, and vice versa—with the restriction that the problem specification contain only a description of the initial state, a description of the goal state, and a description of the allowable actions. Also, in my thesis, I extend the syntax of the language to enhance the parsimony of action descriptions. For example, the description of the Put operator presented in Section 2.2 could be rewritten in the extended language as follows:

Put(p, q)

PRECOND: $p \neq q, p \neq \text{TABLE}, \forall z (\neg \text{On}(z, p)), [q = \text{TABLE} \vee \forall z (\neg \text{On}(z, q))]$

ADD: $\text{On}(p, q)$

DELETE: $\text{On}(p, z)$ for all z such that $z \neq q$

Chapter 2 also shows how the correctness conditions for a plan may be expressed in terms of regression operators, and how regression operators may be constructed from action descriptions. The regression equations presented here, though, tend to produce rather long formulas that may often be reduced to much simpler ones. In my thesis, I show how to add simplification rules to the regression equations to overcome this problem. The thesis also presents a number of theorems on regression operators that do not appear in this report, including a theorem that characterizes the kinds of actions that may be described in the language in terms of the regression operators for those actions.

Chapter 3 of this report shows how a simple planning technique may be derived from a particular theorem of the classical planning problems. The technique combines aspects of means-ends analysis, opportunistic planning, goal protection, goal regression, and constraint formulation and propagation (what Stefik called constraint formulation and propagation corresponds to secondary preconditions and regression in my framework). In my thesis, I expand the technique by incorporating partially ordered (i.e., nonlinear) plans, instantiation variables (i.e., formal objects), and

a variant of hierarchical planning in which abstract operators are constructed dynamically. These devices have the effect of introducing the principle of least-commitment, as they are used to defer search as long as possible. In addition, in my thesis, I remove the various assumptions that are incorporated into the technique presented here, such as the assumption that the initial state is completely known.

Formalization

In formalizing the classical planning problems, we shall draw a distinction between a state of the world and a description of a state. The state of the world is an abstract concept referring to the totality of all that is true of the world and all that is false. To know the state is to be omniscient. A description, on the other hand, is more concrete: it is a collection of facts about the state expressed in some language. Furthermore, a description need not be complete: certain details might be left out, either because they are not known or because they are thought to be unimportant. Hence, there can be more than one state satisfying a given description.

The distinction between states and state descriptions is not new. For example, the distinction was made by McCarthy and Hayes in developing their situation calculus [7]. The reason for emphasizing it here is that it is crucial to the proper characterization of actions. Since actions are assumed to alter the world in a deterministic fashion, performing an action will transform the world from one state to a uniquely determined succedent state. Actions can therefore be characterized as functions mapping states of the world into other states of the world. This is the traditional view of actions, yet, when implementing practical planning systems, many researchers have chosen to characterize actions as functions that map a *description* of one state into a *description* of its successor state. In Section 2.3, we will see that there appear to be actions for which the description of the succedent state would have to be infinite to reflect all of the state changes in their entirety. This is unacceptable from a practical standpoint. Hence, systems that treat actions as functions on state descriptions must necessarily limit the range of problems they can solve. None of this is an issue, however, when actions are treated as functions on states.

2.1 FIRST-ORDER LOGIC FORMALIZATION

The formalization of states, state descriptions, and actions that will now be presented is based on first-order logic. First-order logic was chosen because it provides a very general framework for expressing and solving classical planning problems. In this formalization, states are identified with algebraic structures, state descriptions with well-formed formulas, and actions with functions on algebraic structures. An algebraic structure is a complete account of which relations hold among which objects, and thus determines the truth value of every formula in the language. An algebraic structure therefore corresponds to the notion of a state in that both represent the totality of all that is true and all that is false. Well-formed formulas are used to describe facts about algebraic structures; hence, the relationship between algebraic structures and well-formed formulas is identical to the relationship between states and state descriptions. Consequently, it seems natural to equate states with algebraic structures and state descriptions with well-formed formulas. Actions become formally characterized as functions on structures as a consequence of equating states with structures. In keeping with tradition, we will refer to actions in this framework as *operators* so as to distinguish between the formal characterization of an action and the event that actually takes place in the "real world."

Let us consider how a planning problem would be stated, given the above formalization. Initial-state and goal descriptions are both descriptions of states and, hence, are expressed as sets of well-formed formulas. Thus, we will have a set of formulas Γ describing the initial state and a set G describing the goal state. Operators are described in two parts. The first part states the *preconditions* that must be met before the operator can be applied. For example, in many block-stacking problems, a block can be moved only if no other block is on top of it. Preconditions are just state descriptions and, hence, are expressed as a set of well-formed formulas π .

The second part of an operator description is a description of a function on algebraic structures. This function defines how the operator affects the state of the world when it is applied. Unfortunately, there is no standard way of expressing functions on structures, as they are not an integral part of first-order logic. An appropriate language for specifying operators must therefore

be developed. Before considering how to construct such a language, we need to examine the notion of a structure more closely. An algebraic structure consists of the following elements:

- (1) A nonempty set (class) of objects D called the *domain* of the structure.
- (2) An n -ary relation r on D (i.e., a set-theoretic relation with n arguments whose components are elements of D) for every n -ary relation symbol R .
- (3) An n -ary function f on D for every n -ary function symbol F .
- (4) A distinguished object c in D for every constant symbol C .

The relation/function/object associated with symbol $R/F/C$ is called the *interpretation* of $R/F/C$. As an example, suppose that we have a blocks world consisting of a *TABLE* and three blocks A , B , and C , where blocks A and B are resting on the *TABLE* and block C is stacked on top of block A . Suppose, further, that our language for talking about this world has four constant symbols, A , B , C , and *TABLE*, corresponding to the objects in the world, and one relation symbol *On*, where $\text{On}(x, y)$ means that x is on top of y . Then the structure representing this world would have $\{A, B, C, \text{TABLE}\}$ as its domain, A as the interpretation of A , B as the interpretation of B , C as the interpretation of C , *TABLE* as the interpretation of *TABLE*, and $\{(A, \text{TABLE}), (B, \text{TABLE}), (C, A)\}$ as the interpretation of *On*. Viewed semantically, x is on top of y if and only if the ordered pair $\langle x, y \rangle$ appears in the interpretation of *On*.

To arrive at a practical way of specifying functions on structures, we shall place a number of restrictions on the kinds of functions that may be defined. The first restriction is that a function may not alter the domain of a structure. That is, if \mathcal{M} is a structure and f is a function on structures, then the domain of $f(\mathcal{M})$ is identical to the domain of \mathcal{M} . This restriction is of concern only when we wish to describe the effects of an action that creates or destroys objects in the world. An example of such an action would be the *GENSYM* function in LISP, which creates new LISP atoms. The difficulty here is that the restriction prevents us from modeling the creation and destruction of objects by adding and deleting elements of the domain. However, we

can obtain the same effect by introducing a unary relation, say U , where $U(x)$ is true if and only if x "actually" exists. The domain of the structure would include all objects that could possibly exist; objects would be "created" and "destroyed" by modifying the interpretation of U . Note that this is precisely how GENSYM is implemented in a real computer: GENSYM does not create LISP atoms "out of thin air," but rather it locates an area of unused memory and claims it for use as a new atom. Clearly, the restriction that an operator must preserve the domain of a structure does not affect the kinds of behavior that may be considered; it only influences the way in which the behavior is simulated.

The second restriction is that a function on structures may not alter the language used to describe the world. That is, relation, function, and constant symbols may neither be introduced nor eliminated by an operator. This restriction is implicit in all work done in planning to date. It has never been stated explicitly, since it is hard to imagine a situation in which altering the language would make any sense. Yet, if one really wanted to, one could obtain the effect of modifying the language by introducing relations, functions and constants as objects in the domain (axiomatic set theory [13] provides a convenient way of doing this) and then "creating" and "destroying" them in a manner similar to that described in the preceding paragraph.

The motivation for this second restriction is that it allows a function on structures to be decomposed into a collection of functions—one function for each relation symbol, function symbol, and constant symbol. Each function in the collection defines the interpretation of the corresponding symbol, in the succedent state, in terms of the state of the world that existed prior to the application of the operator. In other words, if f_S is the function corresponding to symbol S and if M is the structure defining the current state of the world, then the interpretation of S in the succedent state is given by $f_S(M)$.

To provide a way of specifying these functions, let us introduce our third and final restriction: each function must be representable as a well-formed formula. That is, each function f_S corresponding to symbol S is defined by a well-formed formula φ_S such that

- (1) For each n -ary relation symbol R , $R(x_1, \dots, x_n)$ is true in the succedent state if and only if $\varphi_R(x_1, \dots, x_n)$ was true previously (where x_1, \dots, x_n are the free variables of φ_R)

- (2) For each n -ary function symbol F , $F(x_1, \dots, x_n) = w$ is true in the succedent state if and only if $\varphi_F(x_1, \dots, x_n, w)$ was true previously.
- (3) For each constant symbol C , $C = w$ is true in the succedent state if and only if $\varphi_C(w)$ was true previously.

For example, suppose we have an operator that places block B on top of block C . After this operator is applied, B becomes situated on top of C and every block except B remains where it was. Therefore, $\text{On}(x, y)$ is true after the application of the operator if and only if $(x = B \wedge y = C) \vee (x \neq B \wedge \text{On}(x, y))$ was true previously. In other words, the interpretation of On in the succedent state is the set of ordered pairs $\langle x, y \rangle$ such that $(x = B \wedge y = C) \vee (x \neq B \wedge \text{On}(x, y))$ is true in the current state. If this operator were applied to the blocks world described earlier, where the interpretation of On was $\{\langle A, \text{TABLE} \rangle, \langle B, \text{TABLE} \rangle, \langle C, A \rangle\}$, the resulting interpretation of On would then be $\{\langle A, \text{TABLE} \rangle, \langle C, A \rangle, \langle B, C \rangle\}$.

With the planning technique discussed later in this paper, it is important to know exactly what modifications an operator makes in a structure to select the appropriate operators for achieving the intended goals. Therefore, we shall express the φ_R 's, φ_F 's and φ_C 's defined above in terms of other formulas that make the modifications explicit and then deal exclusively with these other formulas. For relation symbols, this means expressing each φ_R associated with an operator a in terms of two other formulas, α_R and δ_R , which, respectively, describe the additions to and the deletions from the interpretation of R : if $\alpha_R(x_1, \dots, x_n)$ is true when operator a is applied, the tuple $\langle x_1, \dots, x_n \rangle$ is added to the interpretation of R , and if $\delta_R(x_1, \dots, x_n)$ is true then $\langle x_1, \dots, x_n \rangle$ is deleted from the interpretation of R . For this to make sense, $\alpha_R(x_1, \dots, x_n)$ and $\delta_R(x_1, \dots, x_n)$ cannot be true simultaneously, as we are not requiring that the additions and deletions be performed in any particular order. Given α_R and δ_R , $R(x_1, \dots, x_n)$ is true after operator a is applied if and only if

$$\alpha_R(x_1, \dots, x_n) \vee (\neg \delta_R(x_1, \dots, x_n) \wedge R(x_1, \dots, x_n)) \quad (2.1)$$

was true beforehand. In other words, $\langle x_1, \dots, x_n \rangle$ is in the interpretation of R after applying a if and only if it was added or it was in the interpretation of R beforehand and not deleted. Formula

(2.1) is therefore equivalent to φ_R . Note that appropriate α_R 's and δ_R 's can be found to make (2.1) equivalent to φ_R for any arbitrary φ_R . For example, we can let $\alpha_R(x_1, \dots, x_n)$ be the formula $\varphi_R(x_1, \dots, x_n)$ and $\delta_R(x_1, \dots, x_n)$ be $\neg \varphi_R(x_1, \dots, x_n)$. For efficient problem solving, though, α_R and δ_R should be chosen to reflect the actual additions to and deletions from the interpretation of R . For example, for the block-stacking operator described previously, a suitable $\alpha_{On}(x, y)$ would be $(x = B \wedge y = C)$ and a suitable $\delta_{On}(x, y)$ would be $(x = B \wedge y \neq C)$. Note that $\delta_{On}(x, y)$ cannot be $(x = B)$, since $\alpha_{On}(x, y)$ and $\delta_{On}(x, y)$ are not allowed to be true simultaneously.

The formulas defining the interpretations of the function symbols in the succedent state can be restructured in much the same way as the formulas for relation symbols. In the case of functions, though, we can take advantage of the fact that a function must be defined everywhere, as required by the definition of an algebraic structure. Consequently, $F(x_1, \dots, x_n) = w$ is true after an operator has been applied if and only if the operator changed the value of $F(x_1, \dots, x_n)$ to w or the operator preserved the value of $F(x_1, \dots, x_n)$ and $F(x_1, \dots, x_n) = w$ was true previously. These changes can be described by a single formula μ_F , where $\mu_F(x_1, \dots, x_n, w)$ is true if and only if the value of $F(x_1, \dots, x_n)$ is to be updated to w when the operator is applied. Since functions have unique values, μ_F must have the property that either there is a unique w for which $\mu_F(x_1, \dots, x_n, w)$ is true or there are no w 's for which $\mu_F(x_1, \dots, x_n, w)$ is true. Given such a μ_F , $F(x_1, \dots, x_n) = w$ is true after the operator is applied if and only if

$$\mu_F(x_1, \dots, x_n, w) \vee (\neg \exists v [\mu_F(x_1, \dots, x_n, v)] \wedge F(x_1, \dots, x_n) = w) \quad (2.2a)$$

was true previously; that is, $F(x_1, \dots, x_n) = w$ is true after the operator is applied if and only if either if the value of $F(x_1, \dots, x_n)$ was updated to w , or $F(x_1, \dots, x_n) = w$ was true beforehand and the operator preserved the value of $F(x_1, \dots, x_n)$. Formula (2.2a) is therefore equivalent to φ_F . As with α_R and δ_R , an appropriate μ_F can be found to make (2.2a) equivalent to φ_F for any arbitrary φ_F (e.g., let $\mu_F(x_1, \dots, x_n, w)$ be $\varphi_F(x_1, \dots, x_n, w)$). However, for efficient problem solving, μ_F should be chosen to reflect the actual updates of the interpretation of F . As an example, suppose we wished to model the assignment statement $U \leftarrow V$, where U and V are

program variables. To do so, we could have a function Val mapping program variables to their values, plus an operator that updates $\text{Val}(U)$ to be the value of $\text{Val}(V)$. An appropriate update condition $\mu_{\text{val}}(x, w)$ for this operator would then be $(x = U \wedge w = \text{Val}(V))$.

Constant symbols are handled in exactly the same way as function symbols, since constants are simply functions without arguments. Therefore, $C = w$ is true in the succedent state if and only if

$$\mu_C(w) \vee (\neg \exists v [\mu_C(v)] \wedge C = w) \quad (2.2b)$$

was true previously. Note that Formula (2.2b) is simply a special case of Formula (2.2a).

When dealing with several operators, we will need to distinguish the add, delete, and update conditions of one operator from those of another. This we will do by using superscripts: we will write α_R^a and δ_R^a to mean, respectively, the add and delete conditions defining the interpretation of relation symbol R after operator a is applied, and we will write μ_F^a to mean the update condition defining the interpretation of function symbol F after the application of operator a (likewise for constant symbols). We will also use superscripts to distinguish the preconditions of one operator from those of another. Thus, π^a is the set of preconditions of operator a .

2.2 OPERATOR SCHEMATA

When formulating a planning problem, one quite often encounters groups of operators whose add, delete, and update conditions would be identical given an appropriate substitution of terms. For example, the operator described earlier for stacking block B atop block C has as its add and delete conditions for $\text{On}(x, y)$ the formulas $(x = B \wedge y = C)$ and $(x = B \wedge y \neq C)$, respectively. Similarly, an operator for stacking block A on top of block C would have as its add and delete conditions $(x = A \wedge y = C)$ and $(x = A \wedge y \neq C)$. These formulas are identical except that, wherever B appears in one pair of formulas, A appears in the other. Instead of requiring that each and every operator in such a group be defined separately, we will introduce *operator schemata* so that the group may be defined collectively. Schemata allow one to define parametric classes of

operators by introducing parameters as placeholders for terms in the various formulas that make up an operator definition. A schema is then specialized to a particular operator by substituting the appropriate terms for the parameters. For example, we could define a block-stacking schema with parameters p and q , where p is to be stacked on top of q . The add and delete conditions for $On(x, y)$ in the schema definition would then be $(x = p \wedge y = q)$ and $(x = p \wedge y \neq q)$, respectively. Substituting B for p and C for q yields an operator that stacks block B on top of block C .

It would be useful at this point to introduce a standard notation for defining operators and operator schemata. This notation is illustrated below. A schema definition consists of the name of the schema, a parameter list, and four groups of formulas labeled PRECOND, ADD, DELETE and UPDATE. If the parameter list is empty, the schema defines a single operator.

Name(p_1, \dots, p_m)

PRECOND: $\pi_1(p_1, \dots, p_m), \dots, \pi_n(p_1, \dots, p_m)$

ADD: $R_1(x_1, \dots, x_{n_1})$ for all x_1, \dots, x_{n_1} such that $\alpha_{R_1}(x_1, \dots, x_{n_1}, p_1, \dots, p_m)$

$R_2(x_1, \dots, x_{n_2})$ for all x_1, \dots, x_{n_2} such that $\alpha_{R_2}(x_1, \dots, x_{n_2}, p_1, \dots, p_m)$

...

DELETE: $R_1(x_1, \dots, x_{n_1})$ for all x_1, \dots, x_{n_1} such that $\delta_{R_1}(x_1, \dots, x_{n_1}, p_1, \dots, p_m)$

$R_2(x_1, \dots, x_{n_2})$ for all x_1, \dots, x_{n_2} such that $\delta_{R_2}(x_1, \dots, x_{n_2}, p_1, \dots, p_m)$

...

UPDATE: $F_1(x_1, \dots, x_{n_1}) \leftarrow w$

for all x_1, \dots, x_{n_1}, w such that $\mu_{F_1}(x_1, \dots, x_{n_1}, w, p_1, \dots, p_m)$

$F_2(x_1, \dots, x_{n_2}) \leftarrow w$

for all x_1, \dots, x_{n_2}, w such that $\mu_{F_2}(x_1, \dots, x_{n_2}, w, p_1, \dots, p_m)$

...

The PRECOND group which specifies the precondition of the schema, consists of a set of well-formed formulas $\pi_1(p_1, \dots, p_m), \dots, \pi_n(p_1, \dots, p_m)$ whose free variables are the schema parameters.

The ADD group specifies the add conditions α_R for each relation symbol R . The conditions are

specified by a set of statements of the form

$$\text{"(add) } R(x_1, \dots, x_n) \text{ for all } x_1, \dots, x_n \text{ such that } \alpha_R(x_1, \dots, x_n, p_1, \dots, p_m)\text{"}$$

where the x_i 's are distinct variables and are different from the parameters p_1, \dots, p_m . The x_i 's, together with the parameters, constitute the free variables of α_R . The format of the DELETE group is identical to that of the ADD group. The DELETE group, however, specifies the delete conditions δ_R for each relation symbol R . The UPDATE group specifies the update conditions μ_F and μ_C for each function symbol F and each constant symbol C respectively. These conditions are expressed by a set of statements each of which is of the form

$$\text{"(update) } F(x_1, \dots, x_n) \leftarrow w \text{ for all } x_1, \dots, x_n, w \text{ such that } \mu_F(x_1, \dots, x_n, w, p_1, \dots, p_m)\text{"}$$

for function symbols or, alternatively,

$$\text{"(update) } C \leftarrow w \text{ for all } w \text{ such that } \mu_C(w, p_1, \dots, p_m)\text{"}$$

for constant symbols. As with the ADD and DELETE groups, w and the x_i 's are distinct variables and are different from the parameters.

As an example of what an actual schema might look like, consider the following schema, which defines a class of operators $\text{Put}(p, q)$ for stacking block p on top of q , where q may be another block or the table:

$\text{Put}(p, q)$

PRECOND: $p \neq q, p \neq \text{TABLE}, \forall z (\neg \text{On}(z, p)), [q = \text{TABLE} \vee \forall z (\neg \text{On}(z, q))]$

ADD: $\text{On}(x, y)$ for all x, y such that $(x = p \wedge y = q)$

DELETE: $\text{On}(x, y)$ for all x, y such that $(x = p \wedge y \neq q)$

UPDATE: $A \leftarrow w$ for all w such that *FALSE*

$B \leftarrow w$ for all w such that *FALSE*

$C \leftarrow w$ for all w such that *FALSE*

$\text{TABLE} \leftarrow w$ for all w such that *FALSE*

The precondition states that p and q must be distinct, that p cannot be the table, that no object may be on top of p , and that either q must be the table or no object may be atop q . These are the usual constraints one finds in block-stacking problems.

Since it is often not the case that an operator will modify the interpretation of every symbol in the language, we will introduce the following notational convention: if any α_R , δ_R , μ_F or μ_C is not specified, then we shall take it to be the formula *FALSE*. For example, $\text{Put}(p, q)$, as defined above, does not modify the interpretations of either A , B , C , or TABLE . Therefore, we could define $\text{Put}(p, q)$ more succinctly as follows:

$\text{Put}(p, q)$

PRECOND: $p \neq q, p \neq \text{TABLE}, \forall z (\neg \text{On}(z, p)), [q = \text{TABLE} \vee \forall z (\neg \text{On}(z, q))]$

ADD: $\text{On}(x, y)$ for all x, y such that $(x = p \wedge y = q)$

DELETE: $\text{On}(x, y)$ for all x, y such that $(x = p \wedge y \neq q)$

In essence, the convention is to presume that the interpretation of a symbol is not modified unless specified otherwise. This convention has all the benefits of the "STRIPS assumption" [5]; however, because it is merely a notational convention and we are dealing with functions on states and not functions on state descriptions, it has none of the drawbacks of the STRIPS assumption [16].

We will also adopt as a notational convention that, if no preconditions are given for an operator, then the precondition is taken to be the formula *TRUE*. In other words, we will assume that the operator may be applied in any state.

2.3 VALID PLANS

The statement of a planning problem consists of a set of well-formed formulas Γ describing the initial state of the world, a set of formulas G describing the goals to be achieved, and a set of operator schemata. The object is to find an appropriate sequence of operators (i.e., instantiated schemata) that will transform any structure satisfying Γ into a structure that satisfies G . We shall call such a sequence of operators *a valid plan for achieving G , given Γ* , or simply *a valid plan for*

achieving G when the intended Γ is understood. This section examines the validity conditions in detail and explores ways of testing a plan for validity.

Two conditions must hold for a plan to be valid: first, the preconditions of an operator must be satisfied when that operator is applied; second, the goals must be satisfied after the entire plan has been executed. To state these conditions more precisely, we shall introduce the following definitions. Let ϵ denote the empty sequence—that is, the sequence containing no operators. Let the sequence σ be called a *prefix* of a sequence θ if and only if there exists a sequence γ such that $\theta = \sigma\gamma$ (i.e., θ is equal to the concatenation of σ followed by γ). For example, the prefixes of the sequence $a_1a_2\cdots a_n$ are ϵ , a_1 , a_1a_2 , $a_1a_2a_3$, ..., $a_1a_2\cdots a_n$. Finally, let us write $\Gamma\{\theta\}\varphi$ to mean that, if every formula in the set Γ is true before the sequence of operators θ is applied, then the formula φ will be true after θ is applied. More formally, if we let $a(M)$ denote the structure obtained when operator a is applied to structure M , then

(1) $\Gamma\{\epsilon\}\varphi$ holds if and only if every structure satisfying Γ satisfies φ , and

(2) $\Gamma\{a_1a_2\cdots a_n\}\varphi$ holds if and only if $a_n \circ a_{n-1} \circ \cdots \circ a_1(M)$ satisfies φ for every structure M satisfying Γ ,

where " \circ " denotes function composition. Given the above definitions, the validity conditions may be stated as follows: θ is a valid plan for achieving G given Γ if and only if

(1) $\Gamma\{\theta\}g$ holds for all formulas $g \in G$, and

(2) For every prefix σa of θ , $\Gamma\{\sigma\}\pi_i$ holds for every formula $\pi_i \in \pi^a$, where a is an operator and π^a is the set of preconditions of a .

Unfortunately, it is usually not possible to apply the definition of $\Gamma\{\theta\}\varphi$ directly when testing a plan for validity, as Γ may have an infinite number of models. What we need to do, therefore, is restate the definition of $\Gamma\{\theta\}\varphi$ in terms of theorem proving, so that we may then prove the validity of a plan without having to consider the models of Γ .

Progression Operators

We will consider two possible ways in which the definition of $\Gamma\{\theta\}\varphi$ might be restated in terms of theorem proving. The first approach is to find a *progression operator* [9] for each operator a . Progression operators map the conditions that exist before an action is performed into those that exist after its performance. Thus, if a^{+1} is the progression operator for a , then $\Gamma\{\theta\}\varphi$ holds if and only if φ is a theorem of $a^{+1}(\Gamma)$. If progression operators can be found for each operator a , then the definition of $\Gamma\{\theta\}\varphi$ could be restated as follows:

- (1) $\Gamma\{\epsilon\}\varphi$ if and only if φ is a theorem of Γ , and
- (2) $\Gamma\{a_1 a_2 \cdots a_n\}\varphi$ if and only if φ is a theorem of $a_n^{+1} \circ \cdots \circ a_1^{+1}(\Gamma)$.

Unfortunately, progression operators have a major problem: while it is possible to define an appropriate a^{+1} for any operator a , there appear to be operators and finite Γ 's for which $a^{+1}(\Gamma)$ is necessarily infinite. By definition, $a^{+1}(\Gamma)$ must be an axiomatization of the set of postconditions of Γ ; that is, $a^{+1}(\Gamma)$ must axiomatize $\{\varphi \mid \Gamma\{a\}\varphi\}$. We could simply define $a^{+1}(\Gamma)$ to be this set, but this definition is not practical, as the set of postconditions of Γ is infinite: for computational reasons, we would much prefer a finite axiomatization of the postconditions. Unfortunately, there appear to be cases in which the postconditions cannot be axiomatized finitely, even though Γ may be finite. For example, let Γ be the set of formulas

- Q1: $\forall x (s(x) \neq 0)$
- Q2: $\forall x y (s(x) = s(y) \rightarrow x = y)$
- Q3: $\forall x (x = 0 \vee \exists y (s(y) = x))$
- Q4: $\forall x (x + 0 = x)$
- Q5: $\forall x y (x + s(y) = s(x + y))$
- Q6: $\forall x (x \cdot 0 = 0)$
- Q7: $\forall x y (x \cdot s(y) = (x \cdot y) + x)$
- H1: $\forall x (H(x) \leftrightarrow A(x))$

where $\mathcal{A}(x)$ is a formula that does not contain the symbol H , and let a be the operator whose schema is

$$\text{UPDATE: } x + y \leftarrow w \text{ for all } x, y, w \text{ such that } w = 0$$

$$x \cdot y \leftarrow w \text{ for all } x, y, w \text{ such that } w = 0$$

Formulas Q1 through Q7 are essentially the axioms of Peano arithmetic without the induction axioms. Formula H1 defines the unary relation symbol H in terms of 0, s , $+$, and \cdot by means of the formula $\mathcal{A}(x)$, which will be described below. Operator a leaves the interpretations of 0, s , and H unaltered, but redefines $+$ and \cdot to be zero everywhere after a is applied (i.e., $x + y = x \cdot y = 0$ for all x and y in the succedent state). Since $+$ and \cdot would no longer correspond to addition and multiplication after a is applied, it seems plausible that, if $\mathcal{A}(x)$ made heavy use of addition and multiplication, it might not be possible to finitely axiomatize the postconditions involving H . We will now construct an $\mathcal{A}(x)$ that appears to have just this property.

Let us write $s^n(0)$ as shorthand for the n th successor of 0 (i.e., $s^0(0) = 0$, $s^1(0) = s(0)$, $s^2(0) = s(s(0))$, $s^3(0) = s(s(s(0)))$, etc). Then it can be shown [1] that, for any partial recursive function $p : N^k \rightarrow N$ on the natural numbers, there exists a formula $A_p(x_1, \dots, x_k, y)$ involving only 0, s , $+$, and \cdot such that $p(n_1, \dots, n_k) = m$ if and only if $A_p(s^{n_1}(0), \dots, s^{n_k}(0), s^m(0))$ is a theorem of formulas Q1-Q7. The formula A_p is said to *represent* the function p . Furthermore, it can be shown that, if T_1, T_2, \dots is a recursive enumeration of Turing machines, then there exists a partial recursive indicator function $h : N \rightarrow N$ such that $h(n) = 0$ if and only if T_n eventually halts when started on a blank tape. Let T_1, T_2, \dots be a recursive enumeration of Turing machines and let $\mathcal{A}(x)$ be the formula $A_h(x, 0)$, where h is the partial recursive indicator function defined above and $A_h(x, y)$ is a formula representing h . Having defined $\mathcal{A}(x)$ to be the formula $A_h(x, 0)$, we have as a result that $H(s^n(0))$ is a theorem of Γ if and only if T_n halts on a blank tape. Furthermore, since a does not affect the interpretations of 0, s , or H , $H(s^n(0))$ is a postcondition of Γ if and only if $H(s^n(0))$ is a theorem of Γ . Let Γ' be an axiomatization of the postconditions of Γ . Then $H(s^n(0))$ is a theorem of Γ' if and only if T_n halts on a blank tape. Since $+$ and \cdot are zero everywhere after operator a is applied, we can decompose Γ' into an equivalent set of formulas

$\Gamma'_1 \cup \Gamma'_2$, where Γ'_1 is the set

$$\{\forall x y (x + y = 0) \wedge \forall x y (x \cdot y = 0)\}$$

and Γ'_2 is obtained from Γ' by substituting 0 for all terms of the form $t_1 + t_2$ or $t_1 \cdot t_2$ in every formula of Γ' . Thus, s and H do not appear in Γ'_1 , and $+$ and \cdot do not appear in Γ'_2 . Furthermore, the cardinality of Γ'_2 is less than or equal to the cardinality of Γ' . Since $\Gamma'_1 \cup \Gamma'_2$ is equivalent to Γ' , it follows that $H(s^n(0))$ is a theorem of Γ' if and only if $H(s^n(0))$ is a theorem of $\Gamma'_1 \cup \Gamma'_2$. But s and H do not appear in Γ'_1 . Therefore, the formula $H(s^n(0))$ is true in all structures satisfying $\Gamma'_1 \cup \Gamma'_2$ if and only if it is true in all structures satisfying Γ'_2 . Hence, $H(s^n(0))$ is a theorem of Γ' if and only if $H(s^n(0))$ is a theorem of Γ'_2 . Hence, T_n halts on a blank tape if and only if $H(s^n(0))$ is a theorem of Γ'_2 . But $+$ and \cdot do not appear in any formula of Γ'_2 . Therefore, Γ'_2 must axiomatize H by using only 0 and the successor function s . This seems too weak a language, however, for defining the set of Turing machines that halt on blank tapes without effectively enumerating all such Turing machines. Thus, we make the following conjecture:

Conjecture. Γ'_2 is infinite.

If this conjecture is true, Γ' must be infinite since the cardinality of Γ' is greater than or equal to the cardinality of Γ'_2 . Therefore, all axiomatizations of the postconditions of Γ must be infinite; in particular $a^{+1}(\Gamma)$ must be infinite. Although it appears unlikely that the conjecture is false, it has not yet been formally proved.

Regression Operators

The second approach to restating the definition of $\Gamma\{\emptyset\}\varphi$ is essentially the opposite of the first: instead of advancing Γ forward through the plan using progression operators, we will move φ backwards using *regression operators* [9, 16]. This involves finding for each operator a a function a^{-1} mapping formulas into formulas such that φ is true after applying a if and only if $a^{-1}(\varphi)$ was true beforehand; that is, for every structure M , M satisfies $a^{-1}(\varphi)$ if and only if $a(M)$ satisfies φ . If such functions exist then the definition of $\Gamma\{\emptyset\}\varphi$ could be restated as follows:

- (1) $\Gamma\{\epsilon\}\varphi$ if and only if φ is a theorem of Γ , and
- (2) $\Gamma\{a_1 a_2 \dots a_n\}\varphi$ if and only if $a_1^{-1} \circ \dots \circ a_n^{-1}(\varphi)$ is a theorem of Γ .

In general, a regression operator maps a postcondition into the weakest sufficient precondition that must exist before an operator is applied in order for the postcondition is true afterward. In the case of the a^{-1} 's, though, we are insisting that the weakest sufficient precondition must also be a necessary precondition.

Unlike progression, there are no difficulties in computing regressions. To see why this is so, consider the following construction. First, let us augment our language with an additional set of relation, function, and constant symbols, i.e., one new symbol for each existing symbol. We are thereby adding a new relation symbol R' for each existing relation symbol R , a new function symbol F' for each existing function symbol F , and a new constant symbol C' for each existing constant symbol C . The new symbols we will call primed, the old ones nonprimed. The primed symbols will be used to describe the state of the world that exists after operator a is applied, while the nonprimed symbols will describe the state of the world before a is applied. To axiomatize the relationship between the primed and nonprimed symbols, we can make use of Formulas (2.1) and (2.2) discussed in Section 2.1. These formulas define the interpretation of each symbol after an action has been applied in terms of the previous state of the world. Thus, we have the following axioms for each primed symbol:

$$\forall x_1 \dots x_n [R'(x_1, \dots, x_n) \leftrightarrow \alpha_R^a(x_1, \dots, x_n) \vee (\neg \delta_R^a(x_1, \dots, x_n) \wedge R(x_1, \dots, x_n))] \quad (2.3a)$$

$$\forall x_1 \dots x_n w [(F'(x_1, \dots, x_n) = w) \leftrightarrow \mu_F^a(x_1, \dots, x_n, w) \vee (\neg \exists v (\mu_F^a(x_1, \dots, x_n, v)) \wedge F(x_1, \dots, x_n) = w)] \quad (2.3b)$$

$$\forall w [(C' = w) \leftrightarrow \mu_C^a(w) \vee (\neg \exists v [\mu_C^a(v)] \wedge C = w)] \quad (2.3c)$$

The reason this construction is valid is that operators preserve the domains of the structures to which they are applied: if \mathcal{M} is a structure, then the domain of $a(\mathcal{M})$ is precisely the domain of \mathcal{M} . Therefore, we can construct a composite structure whose domain is the domain shared by \mathcal{M} and $a(\mathcal{M})$, and whose relations, functions, and distinguished elements are the combined relations,

functions, and distinguished elements of \mathcal{M} and $a(\mathcal{M})$. To construct a language for this composite structure, we need only add a new set of symbols to the existing language—one new symbol for each existing symbol, just as was done above.

Now suppose φ is a formula that contains only primed symbols and, hence, describes some condition that might hold after operator a has been applied. Using the axioms given above, we can transform φ into an equivalent formula ψ containing only nonprimed symbols. Since ψ is equivalent to φ and contains only nonprimed symbols, it expresses the necessary and sufficient conditions that must exist before a is applied so that φ will be true afterward. Thus, ψ corresponds to $a^{-1}(\varphi)$.

The transformation of φ into an equivalent nonprimed formula can be done in two steps. The first step is to transform φ into an equivalent canonical form in which every atomic subformula of the canonical φ is either of the form $R'(x_1, \dots, x_n)$, $F'(x_1, \dots, x_n) = w$ or $C' = w$ for some collection of variables x_1, \dots, x_n, w . Once in canonical form, φ can be transformed into its nonprimed equivalent by replacing the atomic subformulas of φ with their equivalent nonprimed formulas, as defined in the axioms (2.3). In other words, we replace all occurrences of

$$R'(x_1, \dots, x_n) \text{ with } \alpha_R(x_1, \dots, x_n) \vee (\neg \delta_R(x_1, \dots, x_n) \wedge R(x_1, \dots, x_n))$$

$$F'(x_1, \dots, x_n) = w \text{ with } \mu_F(x_1, \dots, x_n, w) \vee (\neg \exists v(\mu_F(x_1, \dots, x_n, v)) \\ \wedge F(x_1, \dots, x_n) = w)$$

$$C' = w \text{ with } \mu_C(w) \vee (C = w \wedge \forall v \neg \mu_C(v))$$

These substitutions are justified, since we may always substitute a formula for one that is equivalent. To transform φ into its canonical form, we make use of the following theorem of first-order logic: if $\lambda(\tau)$ is a formula containing the term τ , and if x is neither a free variable of $\lambda(\tau)$ nor a bound variable in the scope of τ , then $\lambda(\tau)$ is logically equivalent to $\exists x(\lambda(x) \wedge \tau = x)$. Therefore, we can replace any occurrence of

$$R'(\dots, \tau, \dots) \text{ with } \exists x(R(\dots, x, \dots) \wedge \tau = x) \\ F'(\dots, \tau, \dots) = \varpi \text{ with } \exists x(F(\dots, x, \dots) = \varpi \wedge \tau = x) \\ F'(\dots) = \tau \text{ with } \exists x(F(\dots) = x \wedge \tau = x) \\ C' = \tau \text{ with } \exists x(C = x \wedge \tau = x),$$

where ϖ is an arbitrary term, τ is a term that is not a variable, and x is a variable that appears in neither $R'(\dots, \tau, \dots)$, $F'(\dots, \tau, \dots) = \varpi$, $F'(\dots)$ nor $C = \tau$. To put φ in canonical form, we merely apply these substitutions repeatedly until no further substitutions are possible.

As an example of how a primed formula is transformed into its nonprimed equivalent, suppose we have the $\text{Put}(B, C)$ operator discussed in Section 2.2 and that φ is $\forall u \neg \text{On}'(u, A')$. To transform φ into its canonical form, we merely need to replace $\text{On}'(u, A')$ with $\exists v (\text{On}'(u, v) \wedge A' = v)$. This produces

$$\forall u \neg \exists v [\text{On}'(u, v) \wedge A' = v].$$

With φ in its canonical form, all that remains is to replace the atomic subformulas of φ with their nonprimed equivalents. Recall from the definition of $\text{Put}(B, C)$ that $\alpha_{\text{On}}(x, y)$ is $(x = B \wedge y = C)$ and $\delta_{\text{On}}(x, y)$ is $(x = B \wedge y \neq C)$. Therefore, all occurrences of $\text{On}'(x, y)$ are replaced by

$$(x = B \wedge y = C) \vee [(x \neq B \vee y = C) \wedge \text{On}(x, y)].$$

Also, since $\text{Put}(B, C)$ does not affect the interpretation of A , μ_A is the formula *FALSE*. Hence, all occurrences of $A' = w$ are replaced by $A = w$. These substitutions produce

$$\forall u \neg \exists v [(u = B \wedge v = C) \vee [(u \neq B \vee v = C) \wedge \text{On}(u, v)]] \wedge A = v],$$

which simplifies to

$$A \neq C \wedge \forall u (u = B \vee \neg \text{On}(u, A)).$$

Thus, no block is on top of A after $\text{Put}(B, C)$ has been applied if and only if A and C are distinct blocks, and there were no blocks on top of A before the application of $\text{Put}(B, C)$, except possibly block B .

The above method for transforming nonprimed formulas into their primed equivalents leads to the following recursive definition for a^{-1} . In the ground case, we obtain

$$a^{-1}[R(x_1, \dots, x_n)] \equiv \alpha_R^a(x_1, \dots, x_n) \vee (\neg \delta_R^a(x_1, \dots, x_n) \wedge R(x_1, \dots, x_n)) \quad (2.4a)$$

$$a^{-1}[F(x_1, \dots, x_n) = w] \equiv \mu_F^a(x_1, \dots, x_n, w) \vee [\neg \exists v (\mu_F^a(x_1, \dots, x_n, v)) \wedge F(x_1, \dots, x_n) = w] \quad (2.4b)$$

$$a^{-1}(C = w) \equiv \mu_C^a(w) \vee [\neg \exists v (\mu_C(v)) \wedge C = w], \quad (2.4c)$$

where x_1, \dots, x_n and w are variables. These equations correspond to replacing atomic subformulas with their nonprimed equivalents. The following equations transform atomic formulas into their

canonical forms:

$$a^{-1}[R(\dots, \tau, \dots)] \equiv \exists x (a^{-1}[R(\dots, x, \dots)] \wedge a^{-1}(\tau = x)) \quad (2.4d)$$

$$a^{-1}[F(\dots, \tau, \dots) = \varpi] \equiv \exists x (a^{-1}[F(\dots, x, \dots) = \varpi] \wedge a^{-1}(\tau = x)) \quad (2.4e)$$

$$a^{-1}[F(\dots) = \tau] \equiv \exists x (a^{-1}[F(\dots) = x] \wedge a^{-1}(\tau = x)) \quad (2.4f)$$

$$a^{-1}(C = \tau) \equiv \exists x (a^{-1}(C = x) \wedge a^{-1}(\tau = x)), \quad (2.4g)$$

where ϖ is an arbitrary term, τ is a term that is not a variable, and x is a variable that does not appear in $R(\dots, \tau, \dots)$, $F(\dots, \tau, \dots) = \varpi$, $F(\dots) = \tau$, or $C = \tau$. Finally, we have the following equations, which allow (2.4a–g) to be applied to all atomic subformulas in a formula:

$$a^{-1}(\neg \varphi) \equiv \neg a^{-1}(\varphi) \quad (2.4h)$$

$$a^{-1}(\varphi \wedge \psi) \equiv a^{-1}(\varphi) \wedge a^{-1}(\psi) \quad (2.4i)$$

$$a^{-1}(\varphi \vee \psi) \equiv a^{-1}(\varphi) \vee a^{-1}(\psi) \quad (2.4j)$$

$$a^{-1}(\varphi \rightarrow \psi) \equiv a^{-1}(\varphi) \rightarrow a^{-1}(\psi) \quad (2.4k)$$

$$a^{-1}(\varphi \leftrightarrow \psi) \equiv a^{-1}(\varphi) \leftrightarrow a^{-1}(\psi) \quad (2.4l)$$

$$a^{-1}(\forall x \varphi) \equiv \forall x a^{-1}(\varphi) \quad (2.4m)$$

$$a^{-1}(\exists x \varphi) \equiv \exists x a^{-1}(\varphi) \quad (2.4n)$$

Plan Synthesis

This chapter presents a technique for solving a subclass of the classical planning problems. The first section establishes the fundamental concepts upon which the technique is based. A particular property of the classical planning problems is identified and an example given illustrating how one might exploit this property when synthesizing a plan. Section 3.2 shows how a simple planning technique can be derived from the property, and, in Section 3.3, a detailed example is provided to demonstrate the technique.

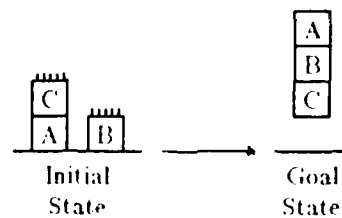
3.1 BASIC CONCEPTS

There are two basic assumptions built into the classical planning problems that can be exploited when a plan is synthesized. The first is that the world can change only as the result of an action. This assumption permits actions to be modeled as state transformations. Furthermore, it forces all plans to have the following property: if some condition is true at one point in a plan but not at an earlier point, then at some point in between there is an operator that causes the condition to become true. This is an important consequence from the point of view of plan synthesis, as it allows one to postulate the existence of operators that cause certain goals to become true. The second assumption is that we are capable of performing only a finite number of actions in a finite amount of time. Consequently, any plan for achieving a particular goal must be finite, as the goal must become true at a definite point in time for it to be achieved. Taken together, these two assumptions imply the following:

Property 3.1. If a condition φ is true at a point p in a sequence of operators but not at an earlier point, then at some point in between there exists an operator that causes φ to become true and φ remains true thereafter until at least point p .

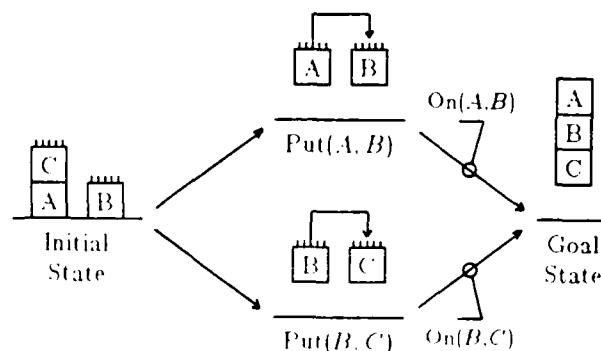
In other words, if some condition is true at one point in a plan but not at an earlier point, then not only must there be an operator somewhere in between that causes the condition to become true, but there must be a final such operator since the number of intervening operators is finite. This combined property turns out to be quite useful during plan synthesis, as we will now demonstrate. A more formal treatment of Property 3.1 appears at the end of this section.

To illustrate how Property 3.1 may be exploited when a plan is being synthesized, let us consider a typical block-stacking problem. Suppose we have the blocks world described in Chapter 2, in which blocks A and B are initially on the *TABLE* and block C is atop block A . Suppose, further, that our goal is to have A on top of B and B on top of C , and that the only operators available are those defined by the Put schema of Section 2.2. The diagram below depicts the initial state and the goal. "bristles" on top of a block signifies that the block is known to be clear (i.e., no other block is on top of it), while a block "floating" above the table signifies that the object supporting the block is not known. The arc from the initial state to the goal signifies that the initial state precedes the goal state in time.

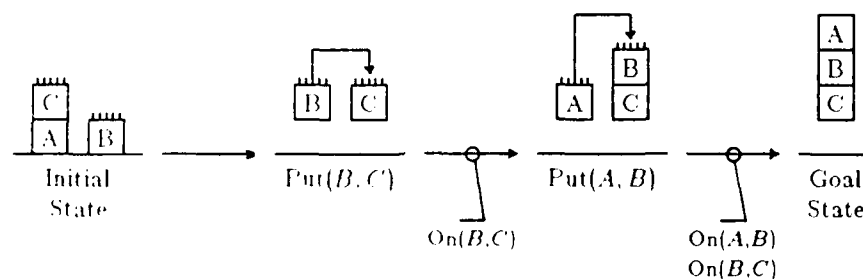


Neither of our goals is satisfied in the initial state; therefore, by Property 3.1 there must be a final point in our plan at which A becomes situated on top of B , and a final point at which B becomes situated on top of C . The only operators available for moving A onto B and B onto C are $\text{Put}(A, B)$ and $\text{Put}(B, C)$, respectively. Hence, there must exist a point at which we apply $\text{Put}(A, B)$, after which A remains on top of B , plus another point at which we apply $\text{Put}(B, C)$, after which B remains on top of C . This is depicted in the diagram below. The conditions that

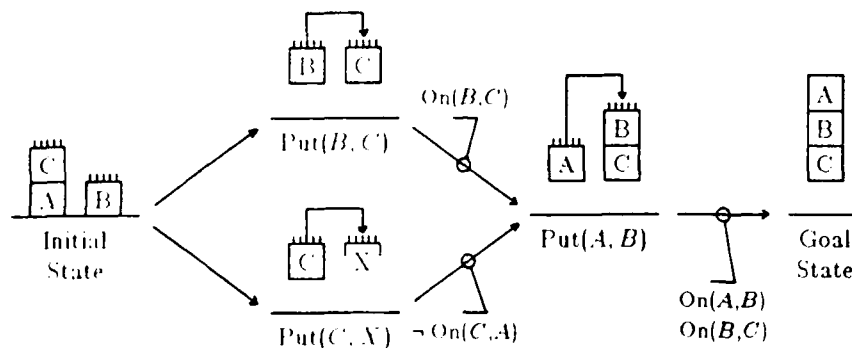
must remain true during particular intervals are identified by labeling the appropriate arcs.



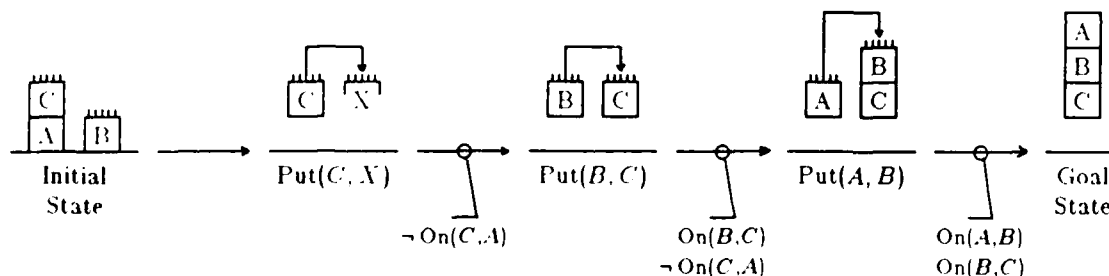
In the final plan, $Put(A, B)$ will come either before $Put(B, C)$ or after $Put(B, C)$. The former case can be ruled out, however, since, with this ordering, the requirement that A remain on top of B after $Put(A, B)$ has been executed contradicts one of the preconditions of $Put(B, C)$, which is that no block be on top of B when $Put(B, C)$ is applied. Therefore, we must perform $Put(A, B)$ after performing $Put(B, C)$.



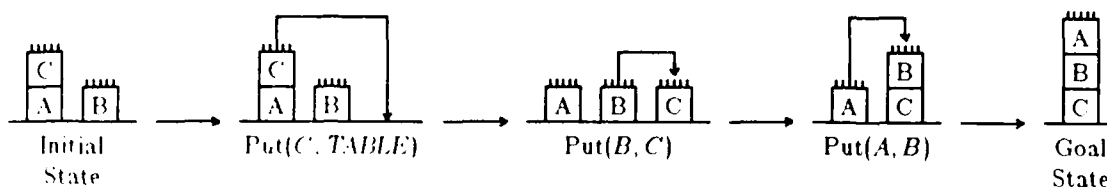
Examining the plan in its current state of development, we find that the goals are now satisfied but that one of the preconditions of $Put(A, B)$ has not been. In particular, C is on top of A in the initial state, which contradicts the requirement that no block be on top of A when $Put(A, B)$ is performed. Therefore, by Property 3.1, there must exist an operator preceding $Put(A, B)$ that causes C to be removed from A and C remains off A thereafter until we perform $Put(A, B)$. The only operators available for removing C from A are those of the form $Put(C, X)$. Hence, there must exist a point preceding $Put(A, B)$ at which we perform $Put(C, X)$ and C remains off A thereafter until we perform $Put(A, B)$. For the moment, let us defer the choice of a particular value for X .



In the final plan, $\text{Put}(C, X)$ will come either before $\text{Put}(B, C)$ or after $\text{Put}(B, C)$. The latter case can be ruled out, however, since, with this ordering, the requirement that B remain on top of C after $\text{Put}(B, C)$ has been executed contradicts one of the preconditions of $\text{Put}(C, X)$, which is that no block be on top of C when $\text{Put}(C, X)$ is applied. Therefore, $\text{Put}(C, X)$ must be applied before $\text{Put}(B, C)$.



If we now examine the plan, we find that every goal and precondition would be satisfied if we were to let X be the *TABLE*. Therefore, let it be so. This gives us the following plan for stacking A atop B and B atop C : put C on the *TABLE*, then put B on top of C and, finally, put A on top of B .



As the foregoing example illustrates, Property 3.1 contributes to the planning process in two ways. First, it establishes a causal connection between the operators in a plan and the

conditions we wish to bring about. This causal linkage permits us to build plans incrementally, introducing operators as needed to satisfy our goals as well as the preconditions of operators previously introduced. The choice of operators is governed by the changes that must be made in the world to bring about the desired conditions; operators that are not essential to constructing a valid plan are not even considered. The result is a tremendous reduction in search compared with that required by an exhaustive search strategy. Furthermore, Property 3.1 does not restrict us to building plans in any particular order, as do forward-chaining and backward-chaining strategies. Instead, operators are inserted *as* needed and *where* needed in an opportunistic fashion (c.f., [6]).

The second way in which Property 3.1 contributes to the planning process is by constraining the placement of operators in a plan. When we insert an operator at some point p in a plan so that a particular condition will be true at some later point q , we are considering the last point p preceding q at which that condition becomes true. We can thus *protect* the condition from point p to point q ; that is, we can assert that the condition must remain true in the interval between p and q . The advantage of protection is that it enables us to detect impossible orderings of operators: if an operator has the precondition φ , it cannot possibly appear at a point in the plan during which $\neg\varphi$ must remain true. Protection therefore contributes to the minimization of search by allowing us to eliminate impossible orderings from consideration. In fact, in the block-stacking example, protection was so effective that search was avoided altogether.

Protection Through the Ages

Historically, the idea of protecting goals and preconditions was first introduced by Sussman [11] and later refined by Waldinger [16], Warren [17, 18], and others. Sussman developed goal protection as a method for detecting faulty plans. As he explains, using a programming metaphor,

... a program is operating correctly, in that it accurately reflects the intent of the programmer, only when each step achieves those goals that the programmer intended it to, and each of those goals remains true at least until the steps which depend upon its being true are run (or the end of the program block if this step is a contributor to the purpose of the program).

Therefore, if, in the course of plan execution, a goal is violated that was intended to remain true, that plan is then faulty and must be "debugged." It is apparent from the foregoing quote that Sussman had in mind something very much like Property 3.1 when he developed his protection mechanism. However, Sussman viewed protection as being intimately tied to the intent of the programmer, whereas here it is seen as arising from a fundamental principle that is independent of intent (of course, in its use, protection does tend to reflect intent). Furthermore, Sussman employed protection only as a means of detecting faulty plans, not as a guide to ordering operators as done here. Had he recognized this use of protection, he probably would not have had to treat the block-stacking problem presented above as an "anomalous situation" requiring special consideration.

Unlike Sussman, Waldinger did employ protection as a guide to ordering operators. However, Waldinger was somewhat overzealous in its application. If a goal or precondition were true in the initial state and not made false by any of the operators currently in the plan, Waldinger's scheme would call for that goal or precondition to be protected without considering the possibility that the goal or precondition might have to be violated and then reestablished in order to solve the overall problem. An example of a problem in which this possibility would have to be considered is the Towers of Hanoi, in which the goal of having the smallest ring on top of the second smallest ring is true in the initial state, but the first ring must be removed from the second so that the other goals can be realized. Waldinger acknowledged that his protection mechanism had drawbacks, but he did not recognize their source. Instead, he proposed a scheme that circumvented the hidden defect by considering goals in various sequences until a solution was obtained. Although the scheme does work, it is terribly inefficient. Furthermore, as Warren points out [18], reordering is unnecessary if we simply avoid protecting goals that are already satisfied.

Like Waldinger, Warren also used protection as a guide to ordering operators. In Warren's approach, though, a goal is protected only when an operator is inserted that makes the goal true. Warren's scheme therefore operates in accordance with Property 3.1.

Strengthening Property 3.1

It turns out that Property 3.1 is too weak for solving arbitrary planning problems. While it works fine for problems in which the effects of an action are independent of the state in which the action was performed, as in the blocks world, it neglects an important case that must be considered when the effects of an action depend on the state of the world at the time the action was performed. Taking this second case into account, we obtain the theorem stated below. This theorem says that a condition is true after a sequence of operators has been executed *if and only if* (1) there exists an operator at some point in the sequence that causes the condition to become true, and the condition remains true thereafter, or (2) the condition is true initially and never becomes false. Therefore, during plan synthesis, not only must we consider incorporating operators to cause a goal or precondition to become true (Clause 1), but we must also consider the possibility of incorporating operators to prevent a goal or precondition from becoming false if it is true initially (Clause 2). Property 3.1 merely provides a set of *sufficient conditions* for Clause (1) to hold. The theorem further tells us that a planning technique is fully general if and only if it takes these two possibilities into account, as a goal or precondition cannot be satisfied otherwise.

Theorem 3.2. Let φ be a formula, Γ be a set of formulas, and θ be a sequence of operators. Then $\Gamma\{\theta\}\varphi$ if and only if one of the following is true:

- (1) There exists a prefix σa of θ , where a is an operator, such that $\Gamma\{\sigma\}\varphi$ is false but $\Gamma\{\sigma a\gamma\}\varphi$ is true for all sequences γ such that $\sigma a\gamma$ is a prefix of θ .
- (2) $\Gamma\{\sigma\}\varphi$ for all prefixes σ of θ .

Proof. First we will show that, if either Clause (1) or Clause (2) holds, then $\Gamma\{\theta\}\varphi$ must hold as well. If σa is a prefix of θ , there exists a γ such that $\sigma a\gamma = \theta$. Therefore, Clause (1) implies $\Gamma\{\theta\}\varphi$. If Clause (2) holds, $\Gamma\{\theta\}\varphi$ follows immediately, since θ is a prefix of itself.

To complete the proof we need to show that, if $\Gamma\{\theta\}\varphi$ holds, then either Clause (1) or Clause (2) holds. This we will do by induction on the length of θ . In the base case, θ is the empty sequence

ϵ . The only prefix of ϵ is ϵ itself; therefore, if $\Gamma\{\theta\}\varphi$ holds for $\theta = \epsilon$, then Clause (2) must hold. For the induction step, let us assume that, for all θ of length less than or equal to n , $\Gamma\{\theta\}\varphi$ implies (1) or (2). Let θ' be a sequence of operators of length $n + 1$, and suppose that $\Gamma\{\theta'\}\varphi$ holds. Let a be an operator and θ'' be a sequence of length n such that $\theta' = \theta''a$. Consider $\Gamma\{\theta''\}\varphi$. Either $\Gamma\{\theta''\}\varphi$ is true or it is false. If it is false, Clause (1) must hold for $\theta = \theta'$ (i.e., consider the case when $\sigma = \theta''$). If $\Gamma\{\theta''\}\varphi$ is true, then, by the induction hypothesis, either Clause (1) or Clause (2) holds for $\theta = \theta''$. If (2) holds for $\theta = \theta''$ then (2) must also hold for $\theta = \theta'$, since we have assumed that $\Gamma\{\theta'\}\varphi$ holds. Likewise, if (1) holds for $\theta = \theta''$, (1) must also hold for $\theta = \theta'$, since, if $\Gamma\{\sigma a \gamma\}\varphi$ is true for all γ such that $\sigma a \gamma$ is a prefix of θ'' , then $\Gamma\{\sigma a \gamma\}\varphi$ must be true for all γ such that $\sigma a \gamma$ is a prefix of θ' . Therefore, if $\Gamma\{\theta''\}\varphi$ holds, either Clause (1) or Clause (2) holds for $\theta = \theta'$. But, as shown previously, if $\Gamma\{\theta''\}\varphi$ does not hold, then Clause (1) must hold for $\theta = \theta'$. Therefore, either Clause (1) or Clause (2) holds for $\theta = \theta'$. Since the choice of θ' was arbitrary, it follows that $\Gamma\{\theta\}\varphi$ implies (1) or (2) for all θ of length $n + 1$. Hence, by induction, $\Gamma\{\theta\}\varphi$ implies (1) or (2) for all θ . \square

Property 3.1 follows as a corollary to Theorem 3.2. Property 3.1 can be stated and proved formally as follows.

Corollary (Property 3.1). Let φ be a formula, Γ a set of formulas, θ a sequence of operators, and τ a prefix of θ . Then the following holds: if $\Gamma\{\theta\}\varphi$ is true but $\Gamma\{\tau\}\varphi$ is false, then there exists a prefix σa of θ such that τ is a prefix of σ , a is an operator, and $\Gamma\{\sigma\}\varphi$ is false, but $\Gamma\{\sigma a \gamma\}\varphi$ is true for all sequences γ such that $\sigma a \gamma$ is a prefix of θ .

Proof. Suppose that $\Gamma\{\theta\}\varphi$ holds but that $\Gamma\{\tau\}\varphi$ does not. Then either Clause (1) or Clause (2) of Theorem 3.2 holds. But Clause (2) cannot hold, since $\Gamma\{\tau\}\varphi$ is false. Therefore, only Clause (1) holds; that is, there exists a prefix σa of θ , where a is an operator, such that $\Gamma\{\sigma\}\varphi$ is false but $\Gamma\{\sigma a \gamma\}\varphi$ is true for all sequences γ such that $\sigma a \gamma$ is a prefix of θ . It remains only to show that τ must be a prefix of σ . Suppose that τ is not a prefix of σ . Since τ and σa are both prefixes of θ , this implies that σa must be a prefix of τ . Therefore, there exists a sequence γ such that $\tau = \sigma a \gamma$.

Therefore, $\Gamma\{\tau\}\varphi$ must be true, since $\Gamma\{\sigma a\gamma\}\varphi$ is true for all sequences γ such that $\sigma a\gamma$ is a prefix of θ . But, by hypothesis, $\Gamma\{\tau\}\varphi$ is false. Contradiction! Therefore, τ must be a prefix of σ . \square

3.2 A SIMPLE PLANNING TECHNIQUE

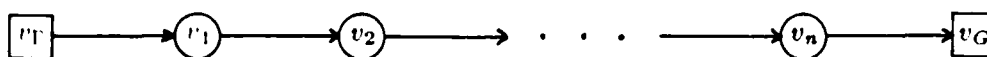
Let us now consider a technique for constructing plans that is based on Theorem 3.2. With this technique, plans are synthesized in much the same way as in the preceding example: we begin with the empty plan (i.e., containing no operators) and add operators until a valid plan is obtained. At each stage in the process, we will have some *current plan*. This plan is analyzed to identify those goals and preconditions not yet satisfied and to determine what additional operators are needed to bring them about. The appropriate operators are then inserted, producing a new current plan and initiating a new cycle of analysis and modification. This process of repeatedly analyzing and modifying the current plan continues until all goals and preconditions have been satisfied. In situations where there are multiple ways of causing a particular goal or precondition to become true, the analysis produces a set of alternative modifications of the current plan. In this case, one of the alternatives must be selected before the plan is modified. However, not all alternatives necessarily lead to solutions, since some ways of effecting one goal or precondition may make it impossible to achieve another. It may thus be necessary to explore a number of alternatives before a valid plan is found.

The technique we shall consider incorporates a number of simplifying assumptions. These assumptions are not essential and, in my thesis [8], I show how they can be lifted to obtain a completely general synthesis technique. The first assumption is that the initial state is completely known. This makes the validity conditions for a plan decidable (in general, they are undecidable). The second assumption is that function symbols and constant symbols do not change interpretation when an operator is applied (i.e., μ_S^a is false for every operator a and every function symbol or constant symbol S). This makes it easier to decompose a complex goal into simpler subgoals. The last assumption is that, for each object x in the world, there is a constant symbol e_x , called

the *standard name* of x , that denotes x in the initial state. Given the preceding assumption, the standard name of x will also denote x at every point in a plan. The reason for this last assumption is that it simplifies the handling of quantifiers.

Representing Plans, Goals, and Protections

To begin, we must establish a representation for plans, goals, and protected conditions. As suggested by the block-stacking example of the previous section, we will represent a plan as a directed acyclic graph, called a *plan graph*, with a single root vertex and a single leaf vertex. The root vertex of a plan graph represents the initial state, while the leaf vertex represents the goal state. The intermediate vertices represent operators. The edges of a plan graph are directed and define a partial ordering of the vertices. From a semantic standpoint, a plan graph asserts that certain operators must appear in the final solution in a certain relative order. Although the representation permits arbitrary partial orders to be specified, we will for the sake of simplicity consider only linear (i.e., totally ordered) plan graphs. An example of a linear plan graph appears below. The diagram uses boxes and circles to distinguish between the root and leaf vertices, on the one hand, and the intermediate vertices on the other.



In our discussion of plan graphs we will adopt the following conventions. We will write v_Γ to denote the root vertex of a plan graph and v_G to denote the leaf vertex, where Γ is the set of formulas describing the initial state and G is the set of formulas defining our goals. If v_1 and v_2 are vertices, we will write $v_1 \triangleright v_2$ to indicate that there is an edge from v_1 to v_2 . The plan graph illustrated above can then be written as $v_\Gamma \triangleright v_1 \triangleright \dots \triangleright v_n \triangleright v_G$. We will say that a vertex v_1 precedes a vertex v_2 , written $v_1 < v_2$, to mean that there is a path from v_1 to v_2 , and we will write $v_1 \leq v_2$ as shorthand for $v_1 < v_2$ or $v_1 = v_2$. Finally, we will say that a formula φ is true at a vertex v to mean either (1) that φ is true in the initial state, if $v = v_\Gamma$, or (2) that φ is true

after execution of the plan, if $v = v_G$, or (3) that φ is true when the operator associated with v is applied, if v is an intermediate vertex. In other words, if our plan graph is $v_T \triangleright v_1 \triangleright \dots \triangleright v_n \triangleright v_G$ and a_i is the operator associated with vertex v_i , then (1) φ is true at v_T if and only if φ is a theorem of Γ , (2) φ is true at v_i if and only if $\Gamma\{a_1 \dots a_{i-1}\}\varphi$, and (3) φ is true at v_G if and only if $\Gamma\{a_1 \dots a_n\}\varphi$.

Protected conditions will be represented as a set P of ordered triples of the form $\langle \varphi, v_1, v_2 \rangle$, where φ is a formula and v_1 and v_2 are vertices such that $v_1 < v_2$. P will be referred to as the *protection set*. In semantic terms, each triple in the protection set is an assertion that a particular formula must remain true over some interval in the final solution. More precisely, if $\langle \varphi, v_1, v_2 \rangle \in P$, then φ must be true at every vertex v in the final plan such that $v_1 < v \leq v_2$. In particular, φ must be true at vertex v_2 . It will be necessary during plan synthesis to consider all protected formulas that must be true at a particular vertex v . Therefore, let us define ρ_v to be this set; that is,

$$\rho_v = \{\varphi \mid \langle \varphi, v_1, v_2 \rangle \in P \text{ and } v_1 < v \leq v_2\} \quad (3.1)$$

Goals and preconditions will be represented as a set A of ordered pairs of the form $\langle \varphi, v \rangle$, where φ is a formula and v is a vertex. We will refer to this set as the *agenda*. From a semantic standpoint, each ordered pair on the agenda is an assertion that a particular formula must be true at a particular vertex in the final plan. In other words, if $\langle \varphi, v_T \rangle \in A$, then φ must be true in the initial state, and, if $\langle \varphi, v \rangle \in A$, where $v \neq v_T$, then one of our goals is to achieve φ at vertex v . The set of all conditions we wish to achieve at a particular vertex v is given by

$$g_v = \{\varphi \mid \langle \varphi, v \rangle \in A\} \quad (3.2)$$

Together, a plan graph, a protection set, and an agenda define a set of constraints that a plan must satisfy to be considered a solution. To synthesize a plan, we take an initial set of constraints defined by an initial plan graph, protection set, and agenda; then, through an appropriate process, we add further constraints until we obtain a complete specification of a plan. At the beginning of the process, our only constraint is for the final plan to achieve every formula g in the goal set

G , given that the initial state is described by the set of formulas Γ . Therefore, the initial plan graph is the graph $v_1 \triangleright v_G$, the initial protection set is empty, and the initial agenda is the set $\{(g, v_G) \mid g \in G\}$. The problem is then to augment each of the three components—the plan graph, the protection set, and the agenda—until the sequence of operators defined by the plan graph satisfies all of the assertions listed in the protection set and the agenda. For convenience, let us refer to the combination of a plan graph, a protection set, and an agenda as a *partial plan*.

The Technique

The synthesis technique we shall consider is an iterative process by which the initial partial plan is incrementally modified until a solution is obtained. The basic loop involves finding a goal on the agenda that would not be satisfied, given the current partial plan, and then modifying the plan so that the goal will be achieved. This process continues until all goals on the agenda have been satisfied. At each step, the current partial plan is modified in such a way that, if all of the goals on the agenda are satisfied, then all of the assertions in the protection set will likewise be satisfied. This guarantees that, once all of the goals have been attained, we will have constructed a valid plan consistent with the protections.

The modifications made of the current partial plan are governed by a set of rules. For every goal that may be expressed in the logic, there is a corresponding rule. Each rule defines a set of alternative modifications for realizing the corresponding goal at the desired point in the final plan. Each set of alternatives covers all possible solutions, so that, if a rule is applicable and if a solution can be obtained from the current partial plan, at least one of the alternatives defined by that rule is guaranteed to lead to a solution. Consequently, the rules may be applied in any order without backtracking and without affecting the final solution. Of course, search is required to explore the alternatives expressed by a rule. As search is fairly well understood, this report will focus on the modification rules and leave open the issue of an appropriate search strategy.

The rules for modifying plans are based in part on Theorem 3.2. According to this theorem, a goal is satisfied at some point p in the final solution if and only if (1) there is an operator that

causes the goal to become true and the goal remains true thereafter until at least point p , or (2) the goal is true initially and never becomes false before point p . This suggests two ways of modifying a partial plan in order to achieve a goal: one is to insert an operator that causes the goal to become true, the other is to prevent the goal from becoming false. There is, however, a third option: since plans are built incrementally, the operator that causes a goal to become true in the final solution may already appear in the current partial plan; therefore, another way of achieving a goal would be to establish the appropriate enabling conditions to allow an existing operator in the plan to cause the goal to become true. These three alternatives are illustrated by the following example.

Suppose we have a world consisting of a briefcase, a dictionary, and a paycheck, each of which may be situated in one of two locations: the home or the office. Operators are available for putting the dictionary and the paycheck into the briefcase and for taking them out, as well as for carrying the briefcase between the two locations. Initially, the briefcase, the dictionary, and the paycheck are at home, and the paycheck is in the briefcase but the dictionary is not. The goal is to have the briefcase and the dictionary at the office, but the paycheck at home. We begin the synthesis process with the empty plan. Let us first consider the goal of having the briefcase at work. Since this goal is not true initially, we must have an operator in our final plan that causes the goal to become true. As the current plan is empty, the only option is to insert the operator that causes the briefcase to be brought to work. Let us next consider the goal of having the dictionary at work. This goal is not satisfied, given the current plan of bringing the briefcase to work. However, if we were to put the dictionary into the briefcase before leaving home, the dictionary would be brought to the office as a side effect. In this case, the operator that causes the dictionary to be at the office (i.e., bringing the briefcase to work) already appears in the plan and an additional operator is inserted to establish the appropriate enabling condition (i.e., having the dictionary in the briefcase). After making these modifications, we are left with only one more goal to consider, which is to have the paycheck remain at home. Unfortunately, the current plan of putting the dictionary in the briefcase and then bringing the briefcase to the office causes the paycheck to be brought to the office as a side effect. However, if we were to remove the paycheck from the

briefcase before leaving home, we would prevent the paycheck from changing locations. Our goal would then be achieved by virtue of the fact that it would never become false. If we choose to remove the paycheck from the briefcase before we put in the dictionary, then our final plan will be to remove the paycheck from the briefcase, put the dictionary in the briefcase, and bring the briefcase to the office.

The three ways of modifying a partial plan illustrated above cover all possible solution paths. This fact is expressed by the theorem that appears below. This theorem may be paraphrased as follows: a condition φ is true at a point p in the final plan if and only if one of the following conditions holds: (1) there exists an operator in the final plan that already appears in the current plan that causes φ to become true, and φ remains true thereafter until at least point p , (2) there exists an operator in the final plan that does not appear in the current plan that causes φ to become true, and φ remains true thereafter until at least point p , or (3) φ is true in the initial state, and remains true until at least point p .

Theorem 3.3. Let θ be a sequence of operators and θ' an expansion of θ . That is, for an appropriate set of operator sequences $\{\beta_1, \beta_2, \dots\}$, if $\theta = a_1 a_2 \dots a_n$, then $\theta' = \beta_1 a_1 \beta_2 a_2 \dots \beta_n a_n \beta_{n+1}$, and, if $\theta = \epsilon$, then $\theta' = \beta_1$. Let $\sigma_1 = \beta_1$ and $\sigma_i = \beta_1 a_1 \dots \beta_{i-1} a_{i-1} \beta_i$ for $i > 1$. Then $\Gamma\{\theta'\}\varphi$ holds if and only if one of the following is true:

- (1) There exists a σ_i such that $\Gamma\{\sigma_i\}\varphi$ is false, but $\Gamma\{\sigma_i a_i \gamma\}\varphi$ is true for all sequences γ such that $\sigma_i a_i \gamma$ is a prefix of θ' .
- (2) There exists a prefix σa of θ' such that σ is not a σ_i and $\Gamma\{\sigma\}\varphi$ is false, but $\Gamma\{\sigma a \gamma\}\varphi$ is true for all sequences γ such that $\sigma a \gamma$ is a prefix of θ' .
- (3) $\Gamma\{\sigma\}\varphi$ holds for all prefixes σ of θ' .

Proof. The above theorem follows directly from Theorem 3.2, as Clauses (1) and (2) together are equivalent to Clause (1) of Theorem 3.2 and Clause (3) is equivalent to Clause (2) of Theorem 3.2

To modify a partial plan to achieve some goal, we merely have to choose one of the three cases described Theorem 3.3 and assert that it holds with respect to the current partial plan and the final solution. In other words, if $\langle \varphi, v \rangle$ is a goal on the agenda and if φ is not true at vertex v in the current partial plan, then we can (1) assert that the operator associated with some existing vertex $v' < v$ causes φ to become true, and protect φ from v' to v , (2) insert an operator that causes φ to become true, and protect φ up to vertex v , or (3) protect φ from the initial state to vertex v .

To make these assertions, we need to introduce the notion of a *secondary precondition*. A secondary precondition for an operator is a condition that must be true at the time the operator is applied for the operator to have the desired effect. By imposing the appropriate secondary precondition on an operator, we can force that operator to preserve some condition or to cause some condition to become true. For example, in the briefcase example discussed earlier, the act of bringing the briefcase to work causes the dictionary to be brought to work as a side effect only if the dictionary happens to be in the briefcase at the time. Therefore, we can achieve the goal of having the dictionary at the office by requiring that the dictionary be in the briefcase when the briefcase is moved. Similarly, to prevent the paycheck from changing locations, we need only require that the paycheck not be in the briefcase at the time the briefcase is moved. To determine which secondary preconditions are appropriate in any given situation, we need to examine more closely the circumstances under which a condition is preserved or is made true by an operator.

For a condition φ to remain true between two points in a plan, all of the intervening operators must preserve the truth of φ ; that is, if φ is true when each such operator is applied, then φ must be true afterward. In Section 2.3 we saw that φ is true after an operator a is applied if and only if $a^{-1}(\varphi)$ was true just prior to the application. Therefore, a will preserve the truth of φ if and only if $\varphi \rightarrow a^{-1}(\varphi)$ is true when a is applied. Given that, in the final plan, φ will be true when a is applied, any formula \mathbb{P}_φ^a such that $\varphi \rightarrow (\mathbb{P}_\varphi^a \leftrightarrow a^{-1}(\varphi))$ is an appropriate secondary precondition to impose on a to ensure that a will preserve φ in the final plan. This is justified by the following lemma:

Lemma 3.4. Let \mathbb{P}_φ^a be a formula such that $\varphi \rightarrow (\mathbb{P}_\varphi^a \rightarrow a^{-1}(\varphi))$. Then the following holds: if φ is true before a is applied, φ will be true after a is applied if and only if \mathbb{P}_φ^a is true beforehand.

Proof. By hypothesis, if φ is true before a is applied, then \mathbb{P}_φ^a is true before a is applied if and only if $a^{-1}(\varphi)$ is true before a is applied. But φ will be true after a is applied if and only if $a^{-1}(\varphi)$ is true beforehand. Therefore, if φ is true before a is applied, then φ will be true after a is applied if and only if \mathbb{P}_φ^a is true beforehand. \square

Let us now consider the conditions that must hold for an operator to cause a formula to become true. Given that the initial state is known completely,¹ an operator a causes a closed formula φ to become true if and only if φ is false before a is applied and true afterward. In Section 2.3 it was shown that φ will be true after applying a if and only if $a^{-1}(\varphi)$ is true beforehand. Therefore, a causes φ to become true if and only if $\neg\varphi \wedge a^{-1}(\varphi)$ is true when a is applied. Although we would guarantee φ to be true after a is applied and false beforehand if we were to impose $\neg\varphi \wedge a^{-1}(\varphi)$ as a secondary precondition for a , it is sufficient to impose a weaker precondition Σ_φ^a , where Σ_φ^a is any formula such that $\neg\varphi \wedge a^{-1}(\varphi) \rightarrow \Sigma_\varphi^a$ and $\Sigma_\varphi^a \rightarrow a^{-1}(\varphi)$. Σ_φ^a has the property that, if φ is false when a is applied, then a will cause φ to become true if and only if Σ_φ^a is true when a is applied; if both φ and Σ_φ^a are true when a is applied, then a will preserve the truth of φ . Imposing Σ_φ^a as a secondary precondition therefore guarantees that φ will become true if it is false. Σ_φ^a is weaker than $\neg\varphi \wedge a^{-1}(\varphi)$, as it allows the possibility that φ will be true when a is applied. The reason we would want to impose Σ_φ^a instead of $\neg\varphi \wedge a^{-1}(\varphi)$ is that we can often find a formula Σ_φ^a that is much simpler than $\neg\varphi \wedge a^{-1}(\varphi)$ and, consequently, is easier to deal with. The justification for using Σ_φ^a instead of $\neg\varphi \wedge a^{-1}(\varphi)$ is provided by the following theorem, which is analogous to Theorem 3.2 except that it is stated in terms of Σ_φ^a and \mathbb{P}_φ^a .

¹ If the initial state were not completely known, it would be possible for φ to be false before a is applied for some of the worlds satisfying the initial state description and true for others. Therefore, the modifications described here apply only when the initial state is known completely. When the initial state is only partially known, $a^{-1}(\varphi)$ must be asserted as the secondary precondition.

Theorem 3.5. Let φ be a formula not containing free variables, let Γ be a complete description of the initial state of the world, and let θ be a sequence of operators. Then $\Gamma\{\theta\}\varphi$ holds if and only if one of the following holds:

- (1) There exists a prefix σa of θ , where a is an operator, such that $\Gamma\{\sigma\}\Sigma_\varphi^a$ holds and $\Gamma\{\sigma a\gamma\}\mathbb{P}_\varphi^b$ holds for all sequences γ and all operators b such that $\sigma a\gamma b$ is a prefix of θ .
- (2) φ is a theorem of Γ and $\Gamma\{\sigma\}\mathbb{P}_\varphi^a$ holds for all prefixes σa of θ .

Proof. If Clause (1) holds, then $\Gamma\{\sigma a\}\varphi$ holds for an appropriate prefix σa of θ . Furthermore, by induction and by using Lemma 3.4, $\Gamma\{\sigma a\gamma b\}\varphi$ must hold for all sequences γ and all operators b such that $\sigma a\gamma b$ is a prefix of θ . Therefore, (1) implies $\Gamma\{\theta\}\varphi$. If Clause (2) holds, then, by induction, $\Gamma\{\sigma\}\varphi$ holds for all prefixes σ of θ . Therefore, (2) implies $\Gamma\{\theta\}\varphi$. Hence, if either Clause (1) or Clause (2) holds, or if both hold, then $\Gamma\{\theta\}\varphi$ must hold as well.

For the converse, suppose that $\Gamma\{\theta\}\varphi$ holds. Then, by Theorem 3.2, one of the following holds:

- (i) There exists a prefix σa of θ , where a is an operator, such that $\Gamma\{\sigma\}\varphi$ is false but $\Gamma\{\sigma a\gamma\}\varphi$ is true for all sequences γ such that $\sigma a\gamma$ is a prefix of θ .
- (ii) $\Gamma\{\sigma\}\varphi$ for all prefixes σ of θ .

Suppose that (i) holds for a suitable prefix σa of θ . Then $\Gamma\{\sigma\}\varphi$ is false and $\Gamma\{\sigma a\}\varphi$ is true. But $\Gamma\{\sigma a\}\varphi$ is true if and only if $\Gamma\{\sigma\}a^{-1}(\varphi)$ is true. Furthermore, since Γ is a complete description of the initial state of the world and since φ does not contain free variables, $\Gamma\{\sigma\}\varphi$ is false if and only if $\Gamma\{\sigma\}\neg\varphi$ is true. Therefore, $\Gamma\{\sigma\}(\neg\varphi \wedge a^{-1}(\varphi))$ holds. Hence, $\Gamma\{\sigma\}\Sigma_\varphi^a$ holds. Furthermore, (i) implies that both $\Gamma\{\sigma a\tau\}\varphi$ and $\Gamma\{\sigma a\tau b\}\varphi$ hold for all sequences τ and all operators b such that $\sigma a\tau b$ is a prefix of θ . Therefore, $\Gamma\{\sigma a\tau\}(\varphi \wedge a^{-1}(\varphi))$ holds for all τ and b such that $\sigma a\tau b$ is a prefix of θ . Hence, $\Gamma\{\sigma a\tau\}\mathbb{P}_\varphi^b$ holds for all τ and b such that $\sigma a\tau b$ is a prefix of θ . Hence, Clause (1) of the theorem holds.

Suppose that (ii) holds. Then φ is a theorem of Γ , since $\Gamma\{\}\varphi$ holds. Furthermore, both $\Gamma\{\sigma\}\varphi$ and $\Gamma\{\sigma a\}\varphi$ hold for all σ and a such that σa is a prefix of θ . Therefore, $\Gamma\{\sigma\}\mathbb{P}_\varphi^a$ holds for all σ and a such that σa is a prefix of θ . Hence, Clause (2) of the theorem holds.

These two cases give us the following: if $\Gamma\{\theta\}\varphi$ holds, then either Clause (1) holds or Clause (2) holds or both. But, from before, if (1) or (2) hold, $\Gamma\{\theta\}\varphi$ must hold as well. Therefore, $\Gamma\{\theta\}\varphi$ holds if and only if either (1) holds or (2) holds or both. \square

As with Theorem 3.2, Clause (1) of Theorem 3.5 can be broken down into two subcases: one in which the operator that makes φ true if it is false must be added to the current partial plan; the other in which the operator that makes φ true if it is false already appears in the plan. Thus, we have the following corollary to Theorem 3.5:

Corollary 3.6. Let θ be a sequence of operators and θ' an expansion of θ . That is, for an appropriate set of operator sequences $\{\beta_1, \beta_2, \dots\}$, if $\theta = a_1 a_2 \dots a_n$, then $\theta' = \beta_1 a_1 \beta_2 a_2 \dots \beta_n a_n \beta_{n+1}$, and, if $\theta = \epsilon$, then $\theta' = \beta_1$. Let $\sigma_1 = \beta_1$ and $\sigma_i = \beta_1 a_1 \dots \beta_{i-1} a_{i-1} \beta_i$ for $i > 1$. Let Γ be a complete description of the initial state and let φ be a formula containing no free variables. Then $\Gamma\{\theta'\}\varphi$ holds if and only if one of the following is true

- (1) There exists a σ_i such that $\Gamma\{\sigma_i\}\Sigma_\varphi^a$ holds and $\Gamma\{\sigma_i a_i \gamma\}\mathbb{P}_\varphi^b$ holds for all sequences γ and all operators b such that $\sigma_i a_i \gamma b$ is a prefix of θ' .
- (2) There exists a prefix σa of θ such that σ is not a σ_i , $\Gamma\{\sigma\}\Sigma_\varphi^a$ holds, and $\Gamma\{\sigma a \gamma\}\mathbb{P}_\varphi^b$ holds for all sequences γ and all operators b such that $\sigma a \gamma b$ is a prefix of θ .
- (3) φ is a theorem of Γ and $\Gamma\{\sigma\}\mathbb{P}_\varphi^c$ holds for all prefixes σa of θ .

Proof. The above theorem follows directly from Theorem 3.5, as Clauses (1) and (2) together are equivalent to Clause (1) of Theorem 3.5, and Clause (3) is equivalent to Clause (2) of Theorem 3.5. \square

Let us now consider the precise modifications that must be made in the plan graph, the agenda, and the protection set to protect φ from the initial state, to force an existing operator to make φ true if it is false, and to insert a new operator that makes φ true if it is false.

According to Clause (3) of Corollary 3.6, if φ is to remain true from the initial state to vertex v in the final plan, then φ must be true in the initial state, and Π_{φ}^a must be true when operator a is applied for every operator a prior to vertex v . Therefore, the following modifications must be made to protect φ from the initial state to vertex v :

- (1) $\langle \varphi, v_I \rangle$ must be added to the agenda to require that φ be true in the initial state.
- (2) $\langle \Pi_{\varphi}^a, v' \rangle$ must be added to the protection set for each vertex v' such that $v_I \prec v' \prec v$ to guarantee that every operator preceding vertex v will preserve the truth of φ .
- (3) $\langle \varphi, v_I, v \rangle$ must be added to the protection set to assert that φ is protected from the initial state to vertex v .

If any of the foregoing additions contradict existing goals and preconditions, no amount of further modification will lead to a solution. This is because it is impossible to simultaneously achieve contradictory goals and protections. Therefore, we can rule out the option of protecting φ from the initial state to vertex v if φ is not true in the initial state or if requiring that the intervening operators preserve the truth of φ contradicts existing goals and protections. Stated more precisely, φ cannot be protected from the initial state to vertex v if

- (1) $\{\varphi\} \cup \Gamma$ is inconsistent, or
- (2) $\{\varphi, \Pi_{\varphi}^a\} \cup g_{v'} \cup \rho_{v'}$ is inconsistent for any vertex v' such that $v_I \prec v' \prec v$,

where $\rho_{v'}$ is the current set of protected conditions that must be true at v' , as defined in Equation 3.1, and $g_{v'}$ is the set of goals currently on the agenda that must be achieved at vertex v' , as defined in Equation 3.2. Note that there is no danger in not detecting these inconsistencies if they are present, as it is impossible to obtain a plan that satisfies inconsistent goals and protections. The only reason for the test is to reduce the search space by eliminating impossible solution paths from consideration. This is fortunate, as the test itself is only partially decidable; that is, while it is always possible to detect inconsistencies if they are present, it is not generally possible to confirm their absence if they are not present. Consequently, detecting an inconsistency requires

an unbounded amount of computation. As the only reason for the test is to prune the search space, spending too much time on it can be worse than not performing the test at all. The compromise is to balance amount of computation spent eliminating alternatives against the amount of computation saved in searching a smaller space, in effect limiting the range of inconsistencies that can be detected. The optimum balance, though, is highly dependent on the problem being considered, so it is hard to make any general statements about where the optimum lies.

Let us next consider how to force an existing operator in the plan to cause φ to become true if it is false, and how then to protect φ up to vertex v . If the existing operator $a_{v'}$ is associated with vertex v' , then, by Clause (1) of Corollary 3.6, $\Sigma_{\varphi}^{a_{v'}}$ must be true when $a_{v'}$ is applied, and \mathbb{P}_{φ}^a must be true when operator a is applied for every operator between v' and v . Therefore, the following modifications have to be made to force the operator associated with vertex v' to guarantee that φ will be true and to protect φ up to vertex v :

- (1) $\langle \Sigma_{\varphi}^{a_{v'}}, v' \rangle$ must be added to the agenda guarantee that φ will be true after applying $a_{v'}$.
- (2) $\langle \mathbb{P}_{\varphi}^{a_{v''}}, v'' \rangle$ must be added to the agenda for each vertex v'' such that $v' \prec v'' \prec v$ to guarantee that every operator between v' and v will preserve the truth of φ .
- (3) $\langle \varphi, v', v \rangle$ must be added to the protection set to assert that φ is protected between vertices v' and v .

If these new goals and protections contradict their existing counterparts, it will be impossible to obtain a solution if the modifications are made. Therefore, we can rule out the possibility of forcing $a_{v'}$ to make φ true if it is false, and then protecting φ up to vertex v , if

- (1) $\{\neg \varphi, \Sigma_{\varphi}^{a_{v'}}\} \cup g_{v'} \cup \rho_{v'}$ is inconsistent, or
- (2) $\{\varphi, \mathbb{P}_{\varphi}^{a_{v''}}\} \cup \rho_{v''} \cup g_{v''}$ is inconsistent for any v'' such that $v' \prec v'' \prec v$.

As before, these conditions are only partially decidable, so we must balance the amount of computation spent pruning the search space against the amount saved in searching a smaller space.

The third and final way of modifying a partial plan to achieve a goal is to insert a new operator that causes the goal to become true if it is false and then to protect the goal up to the point we wish it to be true. Since we are considering only linear plan graphs, the insertion must preserve linearity. Therefore, the new operator must be inserted between two consecutive vertices v_1 and v_2 in the current plan graph (i.e., there must be an edge from v_1 to v_2 in the current plan graph). This is done by creating a new vertex v' , removing the edge from v_1 to v_2 , and adding two new edges, one from v_1 to v' and the other from v' to v_2 . The new operator $a_{v'}$ is then associated with v' . The modifications of the agenda and the protection set are then very much like those for forcing an existing operator to make φ true if it is false. As with forcing, we must guarantee that $a_{v'}$ will cause φ to become true if it is false and that all of the operators between vertices v' and v will preserve φ , and we must assert that φ is protected between v' and v . However, we must also guarantee that the preconditions of the new operator $a_{v'}$ will be true when the operator is applied, and we must guarantee that $a_{v'}$ will preserve all of the conditions protected between vertices v_1 and v_2 . The former is accomplished by adding the preconditions of $a_{v'}$ to the agenda. For the latter, the set of conditions protected between v_1 and v_2 is given by

$$\begin{aligned} & \{\psi \mid \langle \psi, v_3, v_4 \rangle \in P \text{ and } v_3 \preceq v_1 \text{ and } v_2 \preceq v_4\} \\ &= \{\psi \mid \langle \psi, v_3, v_4 \rangle \in P \text{ and } v_3 \prec v_2 \preceq v_4\} \\ &= \rho_{v_2} \end{aligned}$$

Therefore, $a_{v'}$ must preserve every formula in ρ_{v_2} . Thus, the complete set of modifications of the agenda and the protection set are as follows: if operator $a_{v'}$ is being inserted at a new vertex v' between vertices v_1 and v_2 so that φ will be true at vertex v , then

- (1) $\langle \pi_i, v' \rangle$ must be added to the agenda for every π_i in the set $\pi^{a_{v'}}$ of preconditions of $a_{v'}$ to guarantee that the preconditions will be satisfied when $a_{v'}$ is applied.
- (2) $\langle \Sigma_{\varphi}^{a_{v'}}, v' \rangle$ must be added to the agenda guarantee that φ will be true after $a_{v'}$ is applied.
- (3) $\langle \Pi_{\rho_{v_2}}^{a_{v'}}, v' \rangle$ must be added to the agenda for every formula $\psi \in \rho_{v_2}$ to ensure that $a_{v'}$ will preserve all conditions protected between vertices v_1 and v_2 .

- (4) $\langle \mathbb{P}_{\varphi}^{a_{v''}}, v'' \rangle$ must be added to the agenda for each vertex v'' such that $v_2 \preceq v'' \prec v$ to guarantee that every operator between the new vertex v' and vertex v will preserve the truth of φ .
- (5) $\langle \varphi, v', v \rangle$ must be added to the protection set to assert that φ is protected between vertices v' and v .

If these new goals and protections contradict their existing counterparts, it will be impossible to obtain a solution if the modifications are made. Therefore, we can rule out the possibility of inserting $a_{v'}$ between vertices v_1 and v_2 if

- (1) $\{\mathbb{P}_{\psi}^{a_{v'}} \mid \psi \in \rho_{v_2}\} \cup \{\neg \varphi, \Sigma_{\varphi}^{a_{v'}}\} \cup \pi^{a_{v'}} \cup \rho_{v_2}$ is inconsistent, or
- (2) $\{\varphi, \mathbb{P}_{\varphi}^{a_{v''}}\} \cup \rho_{v''} \cup g_{v''}$ is inconsistent for any v'' such that $v_2 \preceq v'' \prec v$.

As before, these conditions are only partially decidable, so we must balance the amount of computation spent pruning the search space against the amount saved in searching a smaller space.

The Rules

While the three ways of modifying a partial plan, as described above, are valid for any formula φ , we will perform these modifications only for formulas of the forms $R(t_1, \dots, t_n)$ and $\neg R(t_1, \dots, t_n)$, where R is a relation symbol and the t_i 's are ground terms (i.e., terms without variable symbols). Goals containing connectives and/or quantifiers will be decomposed into simpler formulas and, ultimately, into one of the two forms above. The reason for doing this is that it leads to a more efficient planning technique, primarily because it is easier to identify operators that cause atomic formulas to become true or false than it is to identify the appropriate operators for arbitrary formulas.

To construct modification rules for atomic formulas and their negations, we can take advantage of our earlier assumption that we will be dealing only with problems in which constant and function symbols cannot change interpretation. This assumption gives rise to the following secondary

preconditions for atomic formulas and their negations:

$$\begin{aligned} \mathbb{P}_R^a(t_1, \dots, t_n) &\equiv \neg \delta_R^a(t_1, \dots, t_n) \\ \mathbb{P}_{\neg R}^a(t_1, \dots, t_n) &\equiv \neg \alpha_R^a(t_1, \dots, t_n) \\ \Sigma_R^a(t_1, \dots, t_n) &\equiv \alpha_R^a(t_1, \dots, t_n) \\ \Sigma_{\neg R}^a(t_1, \dots, t_n) &\equiv \delta_R^a(t_1, \dots, t_n) \end{aligned}$$

In other words, a preserves the truth of $R(t_1, \dots, t_n)$ if and only if a does not delete $\langle t_1, \dots, t_n \rangle$ from the interpretation of R , a preserves the truth of $\neg R(t_1, \dots, t_n)$ if and only if a does not add $\langle t_1, \dots, t_n \rangle$ to the interpretation of R , a causes $R(t_1, \dots, t_n)$ to become true if it is false if and only if a adds $\langle t_1, \dots, t_n \rangle$ to the interpretation of R , and a causes $\neg R(t_1, \dots, t_n)$ to become true if it is false if and only if a deletes $\langle t_1, \dots, t_n \rangle$ from the interpretation of R .

In stating the modification rules for $R(t_1, \dots, t_n)$ and $\neg R(t_1, \dots, t_n)$, we will treat the case in which R is the symbol '=' separately from the general case. Since we have assumed that no operator can change the interpretation of any constant symbol or function symbol, and since by definition no operator can change the interpretation of '=', $\alpha_{=}^a(t_1, t_2)$ and $\delta_{=}^a(t_1, t_2)$ are both *FALSE*. Therefore, it is impossible to make a goal of the form $t_1 = t_2$ or $t_1 \neq t_2$ true if it is not already true. This gives us the following rule:

Rule 1. If $\langle t_1 = t_2, v \rangle$ or $\langle t_1 \neq t_2, v \rangle$ is an unsatisfied goal on the agenda, no further modification of the current partial plan will lead to a solution. Therefore, a different solution path must be considered.

This rule tells us to abandon the current branch in the search space if a goal of the form $t_1 = t_2$ or $t_1 \neq t_2$ is found to be unsatisfied. If all branches are found to lead to dead ends, a solution does not exist.

When R is not the symbol '=', the following two rules apply. Each of the modifications described in these rules is carried out as discussed previously.

Rule 2. If $\langle R(t_1, \dots, t_n), v \rangle$ is an unsatisfied goal on the agenda and R is not the symbol '=', remove $\langle R(t_1, \dots, t_n), v \rangle$ from the agenda and effect one of the following modifications:

- (1) Protect $R(t_1, \dots, t_n)$ from the initial state to vertex v , if $R(t_1, \dots, t_n)$ is true in the initial state and protecting $R(t_1, \dots, t_n)$ does not contradict existing goals and protections.
- (2) For some vertex v' such that $v_\Gamma \prec v' \prec v$, force the operator associated with vertex v' to cause $R(t_1, \dots, t_n)$ to become true if it is false, and then protect $R(t_1, \dots, t_n)$ up to vertex v , provided neither modification introduces an inconsistency.
- (3) Insert a new operator that causes $R(t_1, \dots, t_n)$ to become true if it is false, at a point preceding vertex v that does not contradict existing goals and protections, and then protect $R(t_1, \dots, t_n)$ up to vertex v .

Rule 3. If $\langle \neg R(t_1, \dots, t_n), v \rangle$ is an unsatisfied goal on the agenda and R is not the symbol '=', remove $\langle \neg R(t_1, \dots, t_n), v \rangle$ from the agenda and effect one of the following modifications:

- (1) Protect $\neg R(t_1, \dots, t_n)$ from the initial state to vertex v , if $\neg R(t_1, \dots, t_n)$ is true in the initial state and protecting $\neg R(t_1, \dots, t_n)$ does not contradict existing goals and protections.
- (2) For some vertex v' such that $v_\Gamma \prec v' \prec v$, force the operator associated with vertex v' to cause $\neg R(t_1, \dots, t_n)$ to become true if it is false, and then protect $\neg R(t_1, \dots, t_n)$ up to vertex v , provided neither modification introduces an inconsistency.
- (3) Insert a new operator that causes $\neg R(t_1, \dots, t_n)$ to become true if it is false, at a point preceding vertex v that does not contradict existing goals and protections, and then protect $\neg R(t_1, \dots, t_n)$ up to vertex v .

Note that it is possible for a situation to arise in which none of the modifications described in Rule 2 is consistent with the existing goals and protections. When this happens, no further modification of the partial plan will lead to a solution, since, in virtue of Corollary 3.6, $R(t_1, \dots, t_n)$ can be achieved *if and only if* one of the modifications described in Rule 2 is consistent with existing goals and preconditions. Therefore, when such an inconsistency is detected, we must abandon the current partial plan and try an alternative solution path. This also applies to Rule 3.

Note also that Rules 2 and 3 call for the unsatisfied goal to be removed from the agenda. This is permitted, as the goal will be satisfied in the final plan if the new assertions are satisfied. The rules that follow also call for the removal of unsatisfied goals, for precisely the same reason, once the appropriate modifications have been made.

The remaining rules are used to decompose complex formulas into simpler ones. To decompose a goal of the form $\varphi \wedge \psi$, we make use of the fact that $\varphi \wedge \psi$ is true at some point in a plan if and only if φ and ψ are both true at that point. This leads to the following rule for conjunctive goals:

Rule 4. If $\langle \varphi \wedge \psi, v \rangle$ is a goal on the agenda, remove $\langle \varphi \wedge \psi, v \rangle$ from the agenda and insert $\langle \varphi, v \rangle$ and $\langle \psi, v \rangle$.

It is recommended that this rule be applied regardless of whether $\varphi \wedge \psi$ is satisfied or not, as φ and ψ may then be considered separately at later stages in the synthesis process.

For disjunctive goals, we can make use of the assumption that the initial state is completely known. As a result, $\varphi \vee \psi$ is true at some point in a plan if and only if either φ or ψ is true at that point, or both are (note that this does not necessarily hold when the initial state is not completely known as $\Gamma\{\emptyset\}(\varphi \vee \psi)$ can be true without either $\Gamma\{\emptyset\}\varphi$ or $\Gamma\{\emptyset\}\psi$ being true). This gives us our fifth rule:

Rule 5. If $\langle \varphi \vee \psi, v \rangle$ is an unsatisfied goal on the agenda, remove $\langle \varphi \vee \psi, v \rangle$ from the agenda and insert EITHER $\langle \varphi, v \rangle$ or $\langle \psi, v \rangle$.

The rule for goals involving implication is merely a special case of the preceding rule, since $\varphi \rightarrow \psi$ is equivalent to $\neg \varphi \vee \psi$.

Rule 6. If $\langle \varphi \rightarrow \psi, v \rangle$ is an unsatisfied goal on the agenda, remove $\langle \varphi \rightarrow \psi, v \rangle$ from the agenda and insert EITHER $\langle \neg \varphi, v \rangle$ or $\langle \psi, v \rangle$.

The rule for goals involving the equivalence connective is obtained from Rules 4 and 5 by making use of the fact that $\varphi \leftrightarrow \psi$ is equivalent to $(\varphi \wedge \psi) \vee (\neg \varphi \wedge \neg \psi)$.

Rule 7. If $\langle \varphi \leftrightarrow \psi, v \rangle$ is an unsatisfied goal on the agenda then remove $\langle \varphi \leftrightarrow \psi, v \rangle$ from the agenda and insert EITHER $\langle \varphi, v \rangle$ and $\langle \psi, v \rangle$ or $\langle \neg \varphi, v \rangle$ and $\langle \neg \psi, v \rangle$.

For quantified goals, we can make use of the assumption that every object in the world has a standard name (i.e., there is a constant symbol denoting that object at every point in a plan). Because of this assumption, if $\{e_1, \dots, e_n\}$ is the set of standard names of all objects, then $\forall x \varphi(x)$ is true if and only if $\varphi(e_i)$ is true for all $e_i \in \{e_1, \dots, e_n\}$, and $\exists x \varphi(x)$ is true if and only if $\varphi(e_i)$ is true for some $e_i \in \{e_1, \dots, e_n\}$. Unsatisfied goals of the form $\forall x \varphi(x)$ are handled by separating the cases for which $\varphi(e_i)$ is false from those for which $\varphi(e_i)$ is true in a manner that permits each false case to be considered individually:

Rule 8. If $\langle \forall x \varphi(x), v \rangle$ is an unsatisfied goal on the agenda and $\varphi(e_i)$ is false at vertex v for each standard name $e_i \in \{e_{i_1}, \dots, e_{i_m}\}$, then remove $\langle \forall x \varphi(x), v \rangle$ from the agenda and insert $\langle \varphi(e_{i_1}), v \rangle, \dots, \langle \varphi(e_{i_m}), v \rangle$ and $\langle \forall x (x = e_{i_1} \vee \dots \vee x = e_{i_m} \vee \varphi(x)), v \rangle$.

An example of the use of Rule 8 may be found in the block-stacking example appearing in Section 3.1. In that example, no block may be on top of block A when $\text{Put}(A, B)$ is performed. However, this requirement is not satisfied, given the plan of placing B on top of C and then A on top of B . This is because C is on top of A both in the initial state and after $\text{Put}(B, C)$. Therefore, we would use Rule 8 to decompose $\forall z \neg \text{On}(z, A)$ into $\neg \text{On}(C, A)$ and $\forall z (z = C \vee \neg \text{On}(z, A))$. The subgoal $\neg \text{On}(C, A)$ would then be achieved by inserting a new operator $\text{Put}(C, X)$ and protecting $\neg \text{On}(C, A)$ in the interval between the $\text{Put}(C, X)$ and the $\text{Put}(A, B)$ operators. The subgoal $\forall z (z = C \vee \neg \text{On}(z, A))$ is serendipitously true and no further action need be taken to achieve it.

For an unsatisfied goal of the form $\exists x \varphi(x)$, we must make $\varphi(e_i)$ true for some standard name e_i . From the standpoint of minimizing the search space, it would be preferable to defer the choice of e_i by introducing a variable as a placeholder for the appropriate e_i and then instantiating this variable at some later point in the synthesis process. The mechanisms needed to handle instantiation variables, however, are beyond the scope of this report and are covered in my thesis

[8]. To keep our planning technique simple, we will explicitly consider each and every choice for e_i .

Rule 9. If $\langle \exists x \varphi(x), v \rangle$ is an unsatisfied goal on the agenda and $\{e_1, \dots, e_n\}$ is the set of standard names of all objects in the world, then remove $\langle \exists x \varphi(x), v \rangle$ from the agenda and insert $\langle \varphi(e_i), v \rangle$ for some $e_i \in \{e_1, \dots, e_n\}$.

The remaining rules deal with negated goals. They are obtained from the previous rules in an obvious fashion by making use of De Morgan's laws and similar theorems of first-order logic.

Rule 10. If $\langle \neg(\varphi \wedge \psi), v \rangle$ is an unsatisfied goal on the agenda, remove $\langle \neg(\varphi \wedge \psi), v \rangle$ from the agenda and insert EITHER $\langle \neg \varphi, v \rangle$ or $\langle \neg \psi, v \rangle$.

Rule 11. If $\langle \neg(\varphi \vee \psi), v \rangle$ is a goal on the agenda (be it satisfied or not), remove $\langle \neg(\varphi \vee \psi), v \rangle$ from the agenda and insert $\langle \neg \varphi, v \rangle$ and $\langle \neg \psi, v \rangle$.

Rule 12. If $\langle \neg(\varphi \rightarrow \psi), v \rangle$ is a goal on the agenda (be it satisfied or not), remove $\langle \neg(\varphi \rightarrow \psi), v \rangle$ from the agenda and insert $\langle \varphi, v \rangle$ and $\langle \neg \psi, v \rangle$.

Rule 13. If $\langle \neg(\varphi \leftrightarrow \psi), v \rangle$ is an unsatisfied goal on the agenda, remove $\langle \neg(\varphi \leftrightarrow \psi), v \rangle$ from the agenda and insert EITHER $\langle \neg \varphi, v \rangle$ and $\langle \psi, v \rangle$ or $\langle \neg \psi, v \rangle$ and $\langle \varphi, v \rangle$.

Rule 14. If $\langle \neg(\forall x \varphi(x)), v \rangle$ is an unsatisfied goal on the agenda and if $\{e_1, \dots, e_n\}$ is the set of standard names of all objects in the world, remove $\langle \neg(\forall x \varphi(x)), v \rangle$ from the agenda and insert $\langle \neg \varphi(e_i), v \rangle$ for some $i, 1 \leq i \leq n$.

Rule 15. If $\langle \neg(\exists x \varphi(x)), v \rangle$ is an unsatisfied goal on the agenda and $\neg \varphi(e_i)$ is false at vertex v for each standard name $e_i \in \{e_{i_1}, \dots, e_{i_m}\}$, remove $\langle \neg(\exists x \varphi(x)), v \rangle$ from the agenda and insert $\langle \neg \varphi(e_{i_1}), v \rangle, \dots, \langle \neg \varphi(e_{i_m}), v \rangle$ and $\langle \forall x (x = e_{i_1} \vee \dots \vee x = e_{i_m} \vee \neg \varphi(x)), v \rangle$.

3.3 AN EXAMPLE

To illustrate how a plan would be synthesized by applying the rules just introduced, let us formulate and solve the briefcase problem discussed earlier. The reader will recall that there are three objects, a briefcase, a dictionary and a paycheck, and two locations, the home and the office. Each object is at one of the two locations; furthermore, the dictionary and the paycheck may be in or out of the briefcase. In our formulation, we will have five constant symbols, B, D, P, H and O , corresponding, respectively, to the briefcase, dictionary, paycheck, home, and office. We will also have two relation symbols, 'At' and 'In'. 'At' is a binary relation such that $At(x, y)$ is true if and only if object x is at location y , and 'In' is a unary relation such that $In(x)$ is true if and only if object x is in the briefcase. Initially, the three objects are at home; the paycheck is in the briefcase but the dictionary is not. Therefore, the initial state description Γ contains the following formulas:

- (1) $e_1 \neq e_2$ for all $e_1, e_2 \in \{B, D, P, H, O\}$ such that e_1 and e_2 are distinct
- (2) $\forall x (x = B \vee x = D \vee x = P \vee x = H \vee x = O)$
- (3) $\forall x y (At(x, y) \leftrightarrow [(x = B \vee x = D \vee x = P) \wedge y = H])$
- (4) $\forall x (In(x) \leftrightarrow x = P).$

The formulas defined in Item (1) assert that B, D, P, H and O represent distinct entities, and the formula in Item (2) asserts that these are the only entities in existence. Formula (3) asserts that the only entities that have locations are B, D and P , and they are all at H . Finally, (4) asserts that the only entity in the briefcase is P .

Our objective is to have the briefcase and the dictionary at the office, and the paycheck at home. Therefore, the goal description consists of the formulas $At(B, O)$, $At(D, O)$ and $At(P, H)$. To achieve these goals, we may put objects into the briefcase, remove objects from the briefcase, and move the briefcase between the two locations. We will therefore have three operator schemata, $PutIn(z)$, $TakeOut(z)$ and $MovB(l)$, corresponding to the three allowable actions. These schemata

are defined as follows:

PutIn(z)

PRECOND: $\exists x (At(z, x) \wedge At(B, x))$

ADD: In(p) for all p such that $p = z$

TakeOut(z)

PRECOND: $\exists x (At(z, x) \wedge At(B, x))$

DELETE: In(p) for all p such that $p = z$

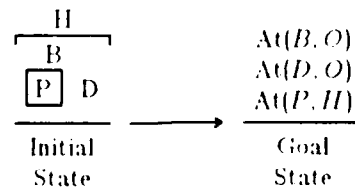
MovB(l)

ADD: At(p, q) for all p, q such that $q = l \wedge (p = B \vee \text{In}(p))$

DELETE: At(p, q) for all p, q such that $q \neq l \wedge (p = B \vee \text{In}(q))$

PutIn(z) causes In(z) to become true and requires as a precondition that z and B be at the same location. TakeOut(z) causes In(z) to become false and also requires as a precondition that z and B be at the same location. MovB(l) causes the briefcase and everything in it to be moved to location l . Unlike PutIn(z) and TakeOut(z), MovB(l) has no precondition and may be applied in any state. If the briefcase and its contents are already at location l , MovB(l) has no effect.

The initial partial plan is illustrated below. The initial state is depicted graphically and the goals are simply listed above the goal vertex. In general, goals on the agenda will be listed above the appropriate vertices and entries in the protection set will be indicated by labeling the appropriate edges.



In this partial plan, At(P, H) is satisfied but At(B, O) and At(D, O) are not; in other words, $\Gamma\{\text{At}(P, H)\}$ holds but $\Gamma\{\text{At}(B, O)\}$ and $\Gamma\{\text{At}(D, O)\}$ do not. Rule 2 can therefore be applied to

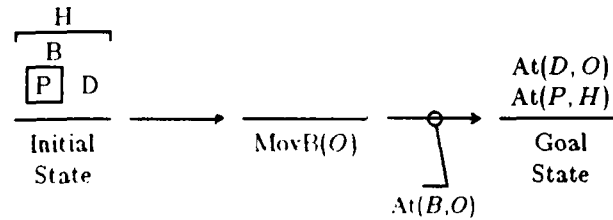
either $At(B, O)$ or $At(D, O)$. It does not matter which we choose to work on first; therefore, let us arbitrarily choose $At(B, O)$. Since $At(B, O)$ is false in the initial state and we are starting with the empty plan, we can rule out protecting $At(B, O)$ from the initial state or forcing an existing operator to make $At(B, O)$ true. Therefore, we have no choice but to insert an operator to make $At(B, O)$ true. $\Sigma_{At(p,q)}^{PutIn(z)}$ and $\Sigma_{At(p,q)}^{TakeOut(z)}$ are both *FALSE* since both $\alpha_{At(p,q)}^{PutIn(z)}$ and $\alpha_{At(p,q)}^{TakeOut(z)}$ are *FALSE*. Hence, neither $PutIn(z)$ nor $TakeOut(z)$ can make $At(B, O)$ true. However,

$$\Sigma_{At(p,q)}^{MovB(l)} \equiv \alpha_{At}^{MovB(l)}(p, q) \equiv [q = l \wedge (p = B \vee In(p))].$$

Therefore,

$$\begin{aligned} \Sigma_{At(B,O)}^{MovB(l)} &\equiv [O = l \wedge (B = B \vee In(B))] \\ &\equiv (O = l). \end{aligned}$$

Hence, the only operator that can make $At(B, O)$ true is $MovB(O)$. Inserting $MovB(O)$ into the plan produces the following plan:



Note that no additions were made to the agenda, since the precondition for $MovB(O)$ is *TRUE* and $\Sigma_{At(B,O)}^{MovB(O)} \equiv (O = O) \equiv \text{TRUE}$ (*TRUE* is always true and thus need not be placed on the agenda). $At(B, O)$, however, was removed from the agenda. This is reflected in the diagram by removing $At(B, O)$ from the goal vertex.

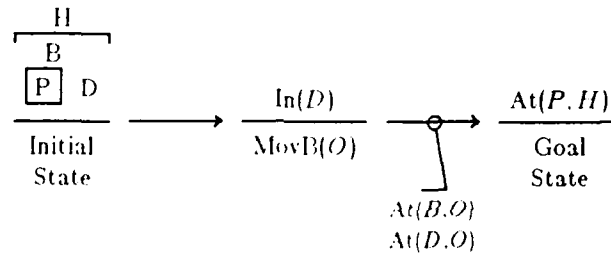
In the above plan, both $At(D, O)$ and $At(P, H)$ are unsatisfied. Choosing to work on $At(D, O)$ first and applying Rule 2, we find that $At(D, O)$ is not true in the initial state; consequently, we must either insert a new operator to make $At(D, O)$ true or force an existing operator to make $At(D, O)$ true. $\Sigma_{At(D,O)}^{PutIn(z)}$ and $\Sigma_{At(D,O)}^{TakeOut(z)}$ are both *FALSE*, but

$$\Sigma_{At(D,O)}^{MovB(l)} \equiv [O = l \wedge (D = B \vee In(D))].$$

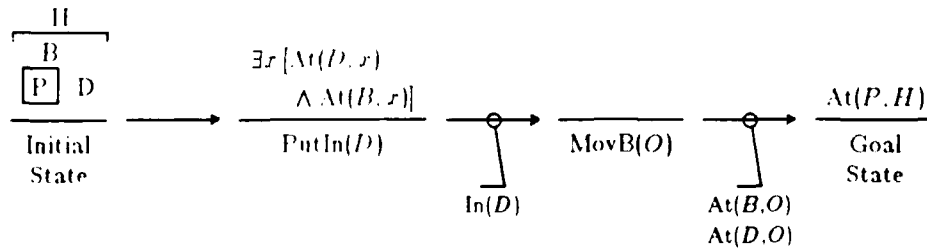
Since $D \neq B$ in the initial state and none of the operators changes the interpretations of either D or B , $\Sigma_{At(D,O)}^{MovB(l)}$ simplifies to

$$\Sigma_{At(D,O)}^{MovB(l)} \equiv (O = l \wedge In(D))$$

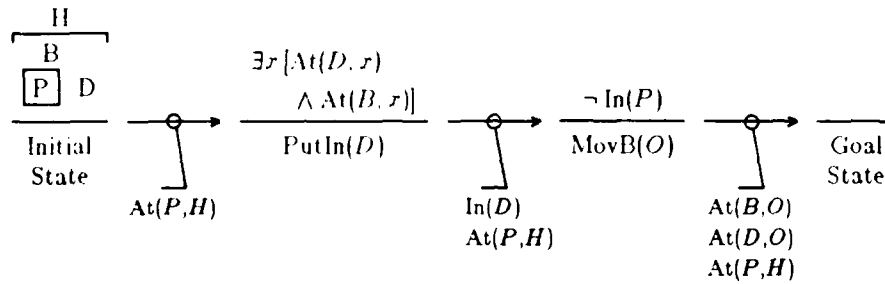
Hence, the only operator that can make $At(D, O)$ true is $MovB(O)$. Therefore, we must either insert a new $MovB(O)$ operator or force the existing $MovB(O)$ operator to cause $At(D, O)$ to become true. Choosing to do the latter, and being prepared to backtrack if this does not work out, we obtain the following partial plan. Note that $In(D)$ is added as a secondary precondition to $MovB(O)$, since $\Sigma_{At(D,O)}^{MovB(O)}$ simplifies to $In(D)$.



In this partial plan, both $In(D)$ and $At(P, H)$ are unsatisfied. Choosing to work on $In(D)$ first and applying Rule 2, we find that $In(D)$ is not true in the initial state and there are no operators preceding $MovB(O)$ in the partial plan. Therefore, our only option is to insert a new operator to make $In(D)$ true. $\Sigma_{In(D)}^{TakeOut(z)}$ and $\Sigma_{In(D)}^{MovB(l)}$ are both *FALSE*, but $\Sigma_{In(D)}^{PutIn(z)} \equiv (D = z)$. Therefore, $PutIn(D)$ is the only operator that can make $In(D)$ true. Inserting $PutIn(D)$ gives us the following partial plan. Note that $\exists x (At(D, x) \wedge At(B, x))$ is placed on the agenda, as it is a precondition for $PutIn(D)$.



In this plan, $\exists x (At(D, x) \wedge At(B, x))$ is satisfied but $At(P, H)$ is not. Applying Rule 2 to $At(P, H)$, we find that either we can protect $At(P, H)$ from the initial state, since it is true initially, or we can insert a new operator $MovB(H)$ with secondary preconditions $In(P)$, since $\Sigma_{At(P, H)}^{MovB(H)} \equiv (H = l \wedge In(P))$ and $\Sigma_{At(P, H)}^{PutIn(D)} \equiv \Sigma_{At(P, H)}^{TakeOut(z)} \equiv FALSE$. Choosing to do the former, and preparing to backtrack if necessary, we obtain the following partial plan:

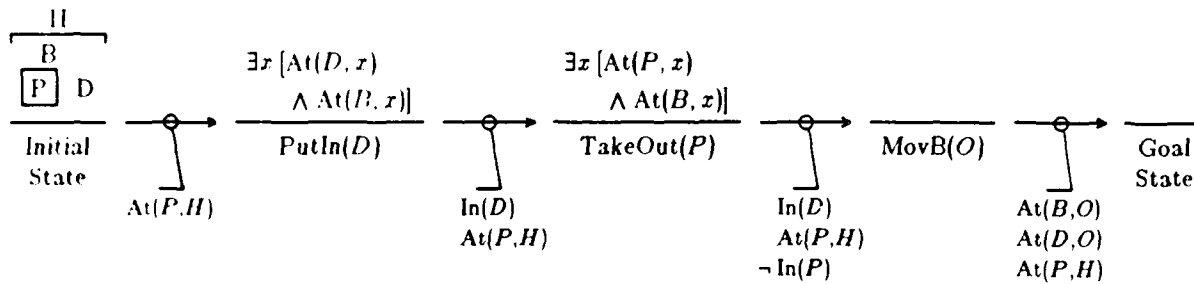


Note that, in protecting $At(P, H)$ from the initial state, $\neg In(P)$ is added as a secondary precondition for $MovB(O)$, since

$$\begin{aligned} \mathbb{P}_{At(P, H)}^{MovB(O)} &\equiv \neg \delta_{At}^{MovB(O)}(P, H) \\ &\equiv \neg [H \neq O \wedge (P = B \vee In(P))] \\ &\equiv \neg In(P) \end{aligned}$$

No other preconditions are imposed on $PutIn(D)$, since $\mathbb{P}_{At(P, H)}^{PutIn(D)} \equiv TRUE$.

At this point, only $\neg In(P)$ is unsatisfied. Applying Rule 3, we find that our only option is to insert a new operator $TakeOut(P)$ either before $PutIn(D)$ or after $PutIn(D)$. Choosing to do the latter, and preparing to backtrack if necessary, we obtain the following partial plan:



All outstanding goals on the agenda are now satisfied. Therefore, the plan just explicated, which consists of putting the dictionary into the briefcase, removing the paycheck from the

briefcase, and then bringing the briefcase to the office, satisfies all of our goals, preconditions, and protections.

Acknowledgments

I would like to thank my advisors, Bob Moore and Gio Wiederhold, for the many long discussions that helped me crystallize my ideas, and for creating an environment that made this research possible. I am especially indebted to Bob Moore for keeping me on track and for providing criticism when needed. Certainly, without Bob's influence, the nature of my work would have been quite different. I have also benefited from discussions with Alfred Aho, David Chapman, Peter Cheeseman, Tom Dean, Mike Georgoff, Amy Lansky, Vladimir Lifshitz, Mike Lowry, John Mohammed, Nils Nilsson, Stan Rosenschein, Marcel Schoppers, Yoav Shoham, Reid Simmons, Albert Visser, Richard Waldinger, and Dave Wilkins.

The research reported herein was supported by the Air Force Office of Scientific Research under Contract No. F49620-82-K-0031, by the Office of Naval Research under Contract Nos. N00014-80-C-0296 and N00014-80-C-0251, and through scholarships from the Natural Sciences and Engineering Research Council Canada and le Fonds F.C.A.C. pour l'aide et le soutien à la recherche, Quebec, Canada. The views and conclusions expressed in this document are those of the author and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the Air Force Office of Scientific Research, the Office of Naval Research, the Natural Sciences and Engineering Research Council Canada, le Fonds F.C.A.C., the U.S. Government, the Quebec Government, or the Canadian Government.

References

- [1] Boolos, G.S. and R.C. Jeffrey, *Computability and Logic*, 2nd edition (Cambridge University Press, Cambridge, England, 1980).
- [2] Chapman, D., "Nonlinear Planning: A Logical Reconstruction," *Proc. IJCAI 9*, University of California at Los Angeles, Los Angeles, California, pp 1022-1024 (August 1985).
- [3] Chapman, D., "Planning for Conjunctive Goals," Tech. Report AI TR-802, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge, Massachusetts (forthcoming).
- [4] Ernst, G.W. and A. Newell, *GPS: A Case Study in Generality and Problem Solving* (Academic Press, New York, New York, 1969).
- [5] Fikes, R.E. and N.J. Nilsson, "STRIPS: A New Approach to the Application of Theorem Proving to Problem Solving," *Artificial Intelligence*, Vol 2, pp 189-208 (1971).
- [6] Hayes-Roth, B., F. Hayes-Roth, S. Rosenschein and S. Cammarata, "Modeling Planning as an Incremental Opportunistic Process," *Proc. IJCAI 6*, Tokyo, Japan, pp 375-383 (August 1979).
- [7] McCarthy, J. and P. Hayes, "Some Philosophical Problems from the Standpoint of Artificial Intelligence," in *Machine Intelligence 4*, Meltzer, B. and D Michie eds., pp 463-502 (Edinburgh University Press, Edinburgh, Scotland, 1969).

- [8] Pednault, E.P.D., *Toward a Mathematical Theory of Plan Synthesis*, Ph.D. thesis, Department of Electrical Engineering, Stanford University, Stanford, California (forthcoming).
- [9] Rosenschein, S.J., "Plan Synthesis: A Logical Perspective," *Proc. IJCAI 7*, University of British Columbia, Vancouver, Canada, pp 331-337 (August 1981).
- [10] Sacerdoti, E.D., "Planning in a Hierarchy of Abstraction Spaces," *Artificial Intelligence*, Vol. 5, No. 2, pp 115-135 (Summer 1974).
- [11] Sacerdoti, E.D., *A Structure for Plans and Behavior* (Elsevier, New York, New York, 1977).
- [12] Stefik, M., "Planning With Constraints (MOLGEN: part 1)," *Artificial Intelligence*, Vol. 16, No. 2, pp 111-140 (May 1981).
- [13] Suppes, P., *Axiomatic Set Theory* (Dover, New York, New York, 1972).
- [14] Sussman, G.J., "A Computational Model of Skill Acquisition," Tech. Report AI TR 197, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge, Massachusetts (August 1973).
- [15] Tate, A., "Project Planning Using a Hierarchic Non-Linear Planner," Research Report No. 25, Department of Artificial Intelligence, University of Edinburgh, Edinburgh, Scotland, (1976).
- [16] Waldinger, R., "Achieving Several Goals Simultaneously," in, *Machine Intelligence 8*, Elcock, E. and D. Michie eds., pp 94-136 (Ellis Horwood, Edinburgh, Scotland, 1977).
- [17] Warren, D.H.D., "WARPLAN: A System for Generating Plans," Memo No. 76, Department of Artificial Intelligence, University of Edinburgh, Edinburgh, Scotland (June 1974).
- [18] Warren, D.H.D., "Generating Conditional Plans and Programs," *Proc. AISB Summer Conference*, University of Edinburgh, Edinburgh, Scotland, pp 344-354 (July 1976).
- [19] Wilkins, D.E., "Domain-Independent Planning: Representation and Plan Generation," *Artificial Intelligence*, Vol. 22, No. 3, pp 269-301 (1984).

END

FILMED

2-86

DTIC